WestVirginiaUniversity
THE RESEARCH REPOSITORY @ WVU

Graduate Theses, Dissertations, and Problem Reports

2019

Immunity-Based Framework for Autonomous Flight in GPS-Challenged Environment

Mohanad Al Nuaimi
malnuaim@mix.wvu.edu

Follow this and additional works at: https://researchrepository.wvu.edu/etd

 Part of the Acoustics, Dynamics, and Controls Commons, Aeronautical Vehicles Commons, Computational Engineering Commons, Controls and Control Theory Commons, Navigation, Guidance, Control and Dynamics Commons, and the Robotics Commons

Recommended Citation

Al Nuaimi, Mohanad, "Immunity-Based Framework for Autonomous Flight in GPS-Challenged Environment" (2019). *Graduate Theses, Dissertations, and Problem Reports*. 4081.
https://researchrepository.wvu.edu/etd/4081

This Dissertation is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Dissertation in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Dissertation has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

Immunity-Based Framework for Autonomous Flight in GPS-Challenged Environment

Mohanad Al Nuaimi

Dissertation submitted to the
Benjamin M. Statler College of Engineering and Mineral Resources at
West Virginia University
In partial fulfillment of the requirements
for the degree of

Doctor of Philosophy
in
Aerospace Engineering

Mario Perhinschi, Ph.D., Chair

Patrick Browning, Ph.D.

Christopher Griffin, Ph.D.

Jason Gross, Ph.D.

Majid Jaridi, Ph. D.

Department of Mechanical and Aerospace Engineering

Morgantown, West Virginia

July 16, 2019

Keywords: Artificial Immune System, GPS-denied Environment, Autonomous Aerial Vehicle.

Copyright 2019 Mohanad Al Nuaimi

ABSTRACT

Immunity-based Framework for Autonomous Flight in GPS-denied Environment

Mohanad Al Nuaimi

In this research, the artificial immune system (AIS) paradigm is used for the development of a conceptual framework for autonomous flight when vehicle position and velocity are not available from direct sources such as the global navigation satellite systems or external landmarks and systems. The AIS is expected to provide corrections of velocity and position estimations that are only based on the outputs of onboard inertial measurement units (IMU). The AIS comprises sets of artificial memory cells that simulate the function of memory T- and B-cells in the biological immune system of vertebrates. The innate immune system uses information about invading antigens and needed antibodies. This information is encoded and sorted by T- and B-cells. The immune system has an adaptive component that can accelerate and intensify the immune response upon subsequent infection with the same antigen. The artificial memory cells attempt to mimic these characteristics for estimation error compensation and are constructed under normal conditions when all sensor systems function accurately, including those providing vehicle position and velocity information. The artificial memory cells consist of two main components: a collection of instantaneous measurements of relevant vehicle features representing the antigen and a set of instantaneous estimation errors or correction features, representing the antibodies. The antigen characterizes the dynamics of the system and is assumed to be correlated with the required corrections of position and velocity estimation or antibodies. When the navigation source is unavailable, the currently measured vehicle features from the onboard sensors are matched against the AIS antigens and the corresponding corrections are extracted and used to adjust the position and velocity estimation algorithm and provide the corrected estimation as actual measurement feedback to the vehicle's control system. The proposed framework is implemented and tested through simulation in two versions: with corrections applied to the output or the input of the estimation scheme. For both approaches, the vehicle feature or antigen sets include increments of body axes components of acceleration and angular rate. The correction feature or antibody sets include vehicle position and velocity and vehicle acceleration adjustments, respectively. The impact on the performance of the proposed methodology produced by essential elements such as path generation method, matching algorithm, feature set, and the IMU grade was investigated. The findings demonstrated that in all cases, the proposed methodology could significantly reduce the accumulation of dead reckoning errors and can become a viable solution in situations where direct accurate measurements and other sources of information are not available. The functionality of the proposed methodology and its promising outcomes were successfully illustrated using the West Virginia University unmanned aerial system simulation environment.

Acknowledgment

I would like to sincerely thank my parents for supporting me throughout my life and showing the value of hard work and that any goal can be accomplished. I would also wish to thank my family and friends for their love and unforgettable support.

Most importantly, I thank my advisor Dr. Mario Perhinschi for his boundless patience, invaluable advice, wisdom, guidance, support, and teachings throughout my education. Also, I thank my other committee members for their guidance and assistance.

Table of Contents

List of Figures viii

List of Tables xii

Nomenclature xiv

1. Introduction 1

1.1. Motivation 1

1.2. Research Objective 2

1.3. Thesis Overview 4

2. Literature Review 5

.2.1 **Autonomous Trajectory Tracking** 5

2.2. Bio-inspired Computational Techniques 6

2.2.1. The Genetic Algorithm 6

2.2.2. Ant-Hill Algorithms 6

2.2.3. Artificial Neural Networks 7

2.2.4. Fuzzy Logic 8

2.2.5. Particle Swarm Optimization 9

2.2.6. The Artificial Immune System 9

2.3. Global Navigation Satellite System 10

2.4. GNSS Vulnerability 11

2.4.1. Uncertainty Sources in GNSS 11

2.4.2. Radio Frequency Interference in GNSS 12

2.5. Inertial Measurement Unit (IMU) 13
3. General Formulation of the AIS-based Framework 15
3.1. The AIS Paradigm 15
3.2. Problem Formulation 16
3.3. Definitions and Notations 17
3.3.1. The Aircraft (or UAV) Reference Frame and Coordinate System 17
3.3.2. Earth Reference Frame and Coordinate System 18
3.3.3. Euler Angles 18
3.3.4. Transformation Matrix 19
3.3.5. Vector Derivative 20
3.3.6. Position Vector 20
3.3.7. Velocity Vector 20
3.3.8. Acceleration Vector 21
3.3.9. Angular Position 21
3.3.10. Angular Velocity Vector 21
3.3.11. Angular Acceleration Vector 22
3.3.12. Actual Values 22
3.3.13. Measured Values 23
3.3.14. Estimated Values 23
3.3.15. AIS-corrected Estimated Values 23

3.3.16. Artificial Memory Cells 23

3.4. AIS-based Framework Architecture 24

3.4.1. The AIS Generation 24

3.4.2. AIS Compensation 28

3.5. AIS Paradigm Challenges 30

4. The WVU UAS Simulation Environment 32

5. Example of Framework Implementation 34

5.1. Baseline Implementation 34

5.1.1. The UAV Features 34

5.1.2. The Acceleration and Angular Rate Measurements 35

5.1.3. The Position and Velocity Estimation 35

5.1.4. The Correction Features 36

5.1.5. The Matching Algorithm 36

5.2. Correction of Estimation Scheme Output 37

5.2.1. AIS Generation 37

5.2.2. AIS-based Estimation Correction 39

5.3. Correction of Estimation Scheme Input 40

5.3.1. AIS Generation 40

5.3.2. AIS-Based Estimation Correction 43

6. Testing and Performance Evaluation 44

6.1. Experimental Design 44
6.2. Correction of Estimation Scheme Output 45
6.3. Correction of Estimation Scheme Input 52
6.4. Effect of Navigation Accuracy on AIS 56
6.5. Path Planning Comparison 59
6.6. IMU Grade Consideration 61
6.7. Affinity Methods Comparison 63
6.8. UAV Features Analysis 64
6.9. Augmentation of Position and Velocity Vertical Components 68

7. Conclusion 71

8. References 73

List of Figures

Figure 1. Immunity cell interactions. 16

Figure 2. The aircraft reference frame. 17

Figure 3. Earth reference frame. 18

Figure 4. Roll, pitch, and yaw rotation axes. 19

Figure 5. Building the AIS (off-line). 25

Figure 6. Estimation scheme and processing corrections for the “output” approach. 26

Figure 7. Estimation scheme and processing corrections for the “input” approach. 26

Figure 8. On-line AIS compensation. 29

Figure 9. Correction of the output of the estimation scheme 29

Figure 10. Correction of the input of the estimation scheme. 30

Figure 11. WVU UAS simulation figure environment mission scenario setup menu. 32

Figure 12. WVU UAS simulation environment–visualization interfaces. 33

Figure 13. Errors model for the IMU components (accelerometer and gyroscope). 33

Figure 14. Immune affinity for antibody extraction. 37

Figure 15. Generating AIS as a collection of artificial memory cells including antibodies of velocity and position. 39

Figure 16.Correction of the estimated position and the velocity components. 40

Figure 17. Closed loop UAV system with AIS correction of position and velocity. 40

Figure 18. Generating AIS as a collection of artificial memory cells including antibodies of acceleration. 42

Figure 19. Correction of the acceleration components. 43

Figure 20. Closed loop UAV system with AIS correction of acceleration. 43

Figure 21. AIS generation trajectory #1. 46

Figure 22.AIS generation trajectory #2. 46

Figure 23. AIS generation trajectory #3. 46

Figure 24. AIS generation trajectory #4. 46

Figure 25. AIS generation trajectory #5. 46

Figure 26. AIS generation trajectory #6. 46

Figure 27. Tracking of AIS generation trajectory under nominal conditions scenario. 48

Figure 28. Tracking of validation trajectory #1 under nominal conditions scenario. 48

Figure 29. Tracking of AIS generation trajectory with estimation scheme only. 48

Figure 30. Tracking of validation trajectory #1 with estimation scheme only. 48

Figure 31. Tracking of AIS generation trajectory with AIS corrected estimation. 49

Figure 32. Tracking of validation trajectory #1 with AIS corrected estimation. 49

Figure 33. Tracking of validation trajectory #2 under nominal conditions scenario. 49

Figure 34. Tracking of validation trajectory #2 with AIS corrected estimation. 49

Figure 35. Tracking of validation trajectory #2 with estimation scheme only. 49

Figure 36. Normalized tracking errors of AIS generation trajectory. 50

Figure 37. Normalized tracking errors of validation trajectory #1. 50

Figure 38. Normalized errors of validation trajectory #2. 50

Figure 39. Nominal generation trajectory. 53

Figure 40. Estimation scheme generation trajectory. 53

Figure 41. AIS generation trajectory. 54

Figure 42. AIS first validation trajectory. 54

Figure 43. AIS second validation trajectory. 54

Figure 44. Normalized tracking errors of generation trajectory. 54

Figure 45. Normalized tracking errors of first validation trajectory. 55

Figure 46. Normalized errors of second validation trajectory. 55

Figure 47. Perfect nominal errors. 58

Figure 48. GPS nominal errors. 58

Figure 49. Effect of GPS on errors. 59

Figure 50. Dubins and clothoid trajectories AIS comparison without estimation. 60

Figure 51. Dubins and clothoid trajectories AIS comparison with estimation. 60

Figure 52. Cross comparison using AMC built with Dubins and clothoid trajectories. 60

Figure 53. Effect of IMU grade on AIS 62

Figure 54. Normalized tracking errors for different affinity methods. 63

Figure 55. Determination of UAV features effect on matching algorithm. 65

Figure 56. UAV features matching accuracy. 65

Figure 57. Weighted mismatching of UAV features. 66

Figure 58. Effect of including the angular acceleration on the matching accuracy. 67

Figure 59. Effect of ω features on the AIS performance. 68

Figure 60. Vertical components compensation. 69

Figure 61. Generation trajectory #1 normalized errors. 70

Figure 62. Generation trajectory #1 normalized errors with estimation. 70

Figure 63. Validation trajectory #1 normalized errors 70

Figure 64. Validation trajectory #1 normalized errors with estimation. 70

List of Tables

Table 1. The percentage improvement of trajectory tracking under the AIS scenario. 51

Table 2. Percentage change in errors for AIS generation trajectory with respect to nominal errors. 51

Table 3. Percentage change in errors for validation trajectory #1 with respect to nominal errors. 51

Table 4. Percentage change in errors for validation trajectory #2 with respect to nominal errors. 51

Table 5. The AIS percentage improvement. 55

Table 6. Percentage change in errors for generation trajectory with respect to nominal errors. .. 56

Table 7. Percentage change in errors for validation trajectory #1 with respect to nominal errors. 56

Table 8. Percentage change in errors for validation trajectory #2 with respect to nominal errors. 56

Table 9. Effect of GPS on errors. 58

Table 10. clothoid errors comparison 61

Table 11. Dubins errors comparison 61

Table 12. IMU grades. 62

Table 13. Trajectory tracking errors of AIS for intermediate and aviation IMU grades. 63

Table 14. Errors comparison of affinity methods with AIS. 64

Table 15. Effect of angular acceleration and angular rate on AIS generation trajectory. 67

Table 16. Effect of angular acceleration and angular rate on AIS validation trajectory. 67

Table 17. Performance metrics comparison without using vertical component sensor augmentation. 69

Table 18. Performance metric comparison with vertical component sensor augmentation. 70

Nomenclature

VARIABLES

Symbol	Description	Units
English		
$\boldsymbol{e}$	Errors	-
$\boldsymbol{\mathcal{F}}$	Correction algorithm	-
$\boldsymbol{D}$	Affinity distance	-
$\boldsymbol{k}$	Time sample	s
$\boldsymbol{L}$	Transformation matrix	-
$\boldsymbol{M_j}$	Memory cell	-
$\boldsymbol{N}$	Population size	-
$\boldsymbol{p}$	Roll rate	rad/s
$\boldsymbol{q}$	Pitch rate	rad/s
$\boldsymbol{r}$	Yaw rate	rad/s
$\vec{\boldsymbol{p}}$	Position vector	m
$\vec{\boldsymbol{v}}$	Velocity vector	m/s
$\vec{\boldsymbol{a}}$	Acceleration vector	m/s^2
Greek		
$\boldsymbol{\alpha}$	Time sample	s
$\boldsymbol{\theta}$	Pitch angle	rad
$\boldsymbol{\varphi}$	Bank angle	rad
$\boldsymbol{\varphi}$	AIS feature	-
$\boldsymbol{\psi}$	Heading angle	rad
$\vec{\boldsymbol{\omega}}$	Angular velocity vector	rad/s
$\dot{\vec{\boldsymbol{\omega}}}$	Angular acceleration vector	rad/s^2
$\sum$	Summation	-
Δ	Difference between the current and the previous values	-

ACRONYMS

Symbol	Description
AI	Artificial Intelligence
AIS	Artificial Immune System
AMC	Artificial Memory Cells
ANN	Artificial Neural Network
CS	Coordinate System
CS_B	Body Coordinate System
CS_E	Earth Coordinate System
DOP	Dilution of Precision
GA	Genetic Algorithm
GPS	Global Positioning System
GNSS	Global Navigation Satellite System
IMU	Inertial Measurement Unit
MEMS	Micro-Electric-Mechanical System
NNSS	Navy Navigation Satellite System
NAVSTAR	Navigation System with Timing and Ranging
PSO	Particle Swarm Optimization
RF	Reference Frame
RF_B	Body Reference Frame
RBF	Radial Basis Functions
TOA	Time of Arrival
UAG	Unmanned Ground Vehicle
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
UD	Upper Matrix and Diagonal Matrix
UKF	Unscented Kalman Filter
WVU	West Virginia University

1. Introduction

1.1. Motivation

The demand for unmanned aerial vehicles (UAVs) has significantly increased due to their benefits and adaptability [1, 2]. UAVs are inexpensive, unmanned, lightweight, versatile, and capable of long endurance, which makes them desirable for use in many fields such as reconnaissance, combat, surveillance, and payload delivery [3]. Safety is a primary concern for all flyable objects. In particular, using the UAVs in an urban areas has numerous limitations due to their safety issues [4]. The autonomous UAVs usually use a global navigation satellite system (GNSS) such as the global positioning system (GPS) to navigate unfamiliar urban and non-urban areas. However, GPS signals can be blocked or severely disturbed, limiting the capability of the system to deliver the required level of availability, accuracy, and reliability of positioning, thus, significantly affecting operational safety [5]. Also, the GPS could be affected by jamming, spoofing, and other technical issues [6]. Introduction of new alternative navigation methods is essential when the GPS is absent, ineffective, or too risky to be used.

Many elements are involved in the operation of autonomous UAVs. Generating the best flyable trajectory of the UAV's path of mission and tracking this trajectory is significantly essential to fulfill the UAV's missions [7]. The trajectory tracking algorithms are expected to follow a commanded trajectory, while minimizing tracking errors. The commanded trajectory path planning can be initiated using a start and finish position, and velocity, waypoints, and obstacles. Al Nuaimi provided a detailed comparison between two methods of path planning [8]. Tracking the path of any moving or flying autonomous vehicle can be achieved using a GNSS [9], which provides feedback of the current position and velocity to a controller that adjusts the speed and the dynamic motion to update the position and velocity [10, 11].

The satellite navigation systems calculate the position of an object relative to known satellites' positions within the Earth, with the satellite signal having the ability to cover a wide area over the globe [12]. The current position and velocity of autonomous vehicles are typically defined using the GNSS. However, reliable alternative solutions that provide similar accuracy and coverage as the satellite navigation systems are needed [13, 14]. Developing an autonomous aerial vehicle that can track the commanded trajectory when the GNSS signal is blocked or inefficient is a challenging objective [15].

The most widely used approach is based on additional information of opportunity, including landmarks that are identified using visual aids and image processing [16]. These methods are inefficient when the mapped area is changing, becomes dark, or is unclear due to environmental effects [16]. Watson and Gross [17] used the factor graph method of Simultaneous Localization and Mapping in conjunction with several robust optimization techniques to evaluate their applicability to robust GNSS data processing. Other methods involve the use of moving and fixed objects with known positions and velocity as a reference to determine the vehicle's position and velocity. The limitation of these methods is that they are expensive and not always feasible [18]. In [19] and [20], Sivaneri and Gross investigated the cooperation between unmanned ground vehicles (UGVs) and UAV for navigation in a GNSS-challenged environment. The focus was on the design of the optimal motion of UGVs to best augment the solution of UAV navigation.

Theoretically, a combination of accelerometers and gyroscopes or inertial measurement unit (IMU) can be used to measure the acceleration and orientation of moving objects. The integral of measured acceleration is used to estimate position and velocity; however, this approach may result in significant biases and large drifts [21]. This dissertation involved the investigation and proposition of a novel approach for correcting position and velocity estimates for autonomous flight vehicles when GPS information and other substitutes are not available based on the artificial immune system (AIS) paradigm.

1.2. Research Objective

The purpose of this research was to investigate the potential of and develop an AIS-based framework for the autonomous tracking of flight trajectories in a GNSS-challenged environment. The developed methodology is implemented and tested through simulation using the West Virginia University (WVU) unmanned aerial system (UAS) simulation environment [22]. The AIS paradigm [23] is inspired by mechanisms of the biological immune system of superior vertebrates which is capable of detecting and counteracting intruding exogenous entities (antigens) while overlooking the self cells [24]. The AIS paradigm exhibits highly robust and adaptive classification and information structuring capabilities, as well as memory and information fusion potential [25].

All these characteristics are beneficial in solving challenging technical problems such as autonomous trajectory tracking when GNSS information is not available. Mechanisms of the biological immune system are mimicked to compensate for the drifting error from integrated acceleration and obtain adequate position and velocity estimates. The general feasibility, applicability conditions, constraints, and benefits of the proposed methodology were analyzed. In this research, a general AIS based framework was formulated to correct the inertial sensor outputs. The AIS classification and memory capabilities were extended and used for autonomous flight control purposes.

The AIS can significantly mitigate the accumulating errors of the inertial sensors output caused by the integration of acceleration components. The AIS can be used not only with known trajectories that were used to build the AIS but also with new different trajectories. The AIS navigation methodology does not require supporting infrastructure, landmarks, or image processing capabilities. The construction of the AIS requires extensive data that can be acquired before the mission at times and locations when vehicle position and velocity information is available.

The AIS paradigm is used for the first time in this context for UAV flight in GPS-challenged environment. The AIS paradigm represents a significant step towards developing a comprehensive and integrated solution for monitoring and controlling aerospace systems, including navigation and trajectory tracking. Note that these sets of data must be representative and complete in defining targeted system operational envelope.

A novel structure is developed and tested extensively for UAV trajectory tracking based on principles of artificial immune systems and used to analyze the effectiveness and performance of the proposed immunity-based framework. Two approaches for the proposed method were developed: 1) correction of the estimation scheme input, and 2) correction of the estimation scheme output. The artificial immune system was built with different trajectories and tested using generation and validation trajectories. The effect of path planning algorithm for the commanded trajectory, model of the affinity method, class of the selected sensors, selected features for the matching algorithm, and scenario when an external source of the vertical components' measurements is available on the proposed method were investigated.

1.3. Thesis Overview

Chapter 1 has provided a comprehensive overview of the introduction and objectives of the dissertation. Chapter 2 presents a literature review of the autonomous trajectory tracking and bio-inspired techniques used to solve it, followed by a discussion of the global navigation satellite system and its vulnerabilities. In addition, Chapter 2 provides a discussion of the inertial measurement unit and the source and modeling of its errors. Chapter 3 provides an overview of the general formulation of the AIS-based framework for autonomous trajectory tracking in GNSS challenged environments. Additionally, Chapter 3 describes the AIS paradigm, the problem formulation, definitions and notations, AIS-based framework architecture, AIS paradigm challenges, and the AIS generation. Chapter 4 provides an overview of the UAV simulation environment used to build and test the AIS framework. Chapter 5 provides an example of using and implementing the proposed AIS framework in two different tactics; one based on correcting the estimation output and the other based on correcting the estimation input. The matching algorithm and affinity techniques are also discussed in Chapter 5. Testing results and their analysis are presented in Chapter 6. Finally, Chapter 7 presents the conclusions of the dissertation based on the findings.

2. Literature Review

2.1. Autonomous Trajectory Tracking

The capability of a system to accomplish tasks and missions without direct input from human operators is known as autonomy [26]. Autonomy implies that the system must possess characteristics of intelligence such as making decisions, performing self-configuration and optimization, sensing and evaluating the system status, and external context [27]. The most critical aspects of autonomous flight are adequate commanded trajectory generation, adjustment and modification, and high-performance control laws for trajectory tracking [28]. Given that following the commanded trajectory is the ultimate objective of autonomous flight, the position, and velocity of the vehicle in the inertial frame are critical. Three large classes of methods have been used individually or as a combination to determine the velocity and position of vehicles: Methods based on on-board sensors that can determine the relative vehicle location based on a known position of the initial point, those based on external sensors such as GNSS, and strategies utilizing on-board sensors capable of detecting or estimating relative vehicle position with respect to landmarks with a priori known locations [2].

The GNSS-based approach has become an effective solution because of its reliability and low cost. However, alternative approaches must be developed to be used as a substitution in the numerous situations when the GNSS is unavailable, not functioning correctly, or undesirable for use. An autonomous trajectory tracking and obstacle avoidance navigation system in GPS denied, and the cluttered environment was developed by Mohta *et al.* [29] using visual sensors. The navigation system consists of a set of integrated modules that work together to allow a quadrotor robot to move from a starting position to a specified location. Sensor fusion was applied to cameras, IMU, and lidar with unscented Kalman filter to determine position and velocity. Software architecture for safe and reliable autonomous navigation of aerial robots in GPS-denied areas was presented by Perez Grau [18] using a six-dimensional approach for localization and state estimation. Visual odometry and Monte Carlo localization were used for motion planning. Dead reckoning represents a class of algorithms that use inertial sensor measurements to obtain integration changes in position or velocity [30]. Zhou *et al* applied the dead-reckoning and discrete Kalman filter with the method of analytic geometry to improve the trajectory tracking accuracy of an unmanned quadrotor [31].

2.2. Bio-inspired Computational Techniques

Bio-inspired techniques result from the transfer of ideas, concepts, mechanisms, or general knowledge from the biological world into technical applications. The development and use of such approaches have remarkably increased over the last few decades in part due to the advent of powerful computers that made the associated substantial computational effort affordable [32]. Before the computers were invented, biological systems solved similar complex problems for a long time, and with the availability of powerful computational tools today, tapping into this source of inspiration is possible and promising.

2.2.1. The Genetic Algorithm

John Holland and colleagues developed the Genetic Algorithm (GA) concept in the 1960s and 1970s [33]. The GA is a biological evolution model based on Charles Darwin's natural selection theory. Studies using GAs have significantly increased in many fields, including adaptive agents in economic theory and design of sophisticated devices such as aircraft turbines and integrated circuits [34]. Nikolos [35] used a combination of a modified breeder GAs incorporated with characteristics of classic ones to create an evolutionary-based framework to design an offline/online path planning for UAVs autonomous navigation. The path planner was used in a three-dimensional environment to calculate the path with curvature in rough terrain. An alternative way of position localization of quadrotor without using GPS or cameras was presented by Faelden *et al*. [36]. The method was dependent on using a transceiver's signal as inputs for the genetic algorithm to locate the quadrotor in x, y, and z-axis. Wilburn *et al.* [37] used a modified GA for gain optimizing of trajectory-tracking controllers for autonomous aircraft which facilitates the investigation of novel control architectures regardless of complexity and dimensionality.

2.2.2. Ant-Hill Algorithms

The ant colony algorithm is a technique for obtaining the optimal path which is compared to an ant's behavior when searching for food. The complex social behavior of ants when navigating in search of food sources using identical traffic paths (or ant streets) has drawn the attention of scientists[38]. Ants are animals with low-resolution vision [39] who can find the shortest routes between the feeding sources and their colony by repeatedly marking their paths with pheromones[40]. Initially, the ants choose random paths to the food source and mark them

with pheromone deposits. The ants who take the shortest path arrive at the food source and back at the nest faster than the others who take longer paths; meaning that a higher amount of pheromone will be deposited on the shortest path as oppossed to the other paths. Because this path has the highest pheromone concentration, other ants will select it, and also deposit more pheromone causing even more concentration. Therefore, the shortest path is the optimal selected path to the food source [38]. The first attempt of the ant algorithm approach was in the early '90s by Dorigo and colleagues [40]. The new approach uses a combination of distributed computation, positive feedback, and a constructive greedy heuristic for stochastic optimization and problem-solving. Dorigo applied his approach to the classical traveling salesman problem with the results showing that the system can rapidly provide adequate solutions. A hybrid improvement strategy for the basic ant colony algorithm model was proposed by Ma and colleagues [41] for UAV optimal trajectory planning in a complex environment. The proposed method shows better performance than the basic ant colony algorithm in convergence speed, solution variation, dynamic convergence behavior, and computational efficiency.

2.2.3. Artificial Neural Networks

One of the unusual sources of inspiration for soft computing techniques is how the human brain works. The functionality of the human brain relies on many specialized cells interacting together within a highly interconnected system called a neural network. The Artificial Neural Network (ANN) is a learning-based computing technique based on information processing inspired by the brain's biological neural network [42]. The ANN attempts to mimic the way information is processed by the brain. The first attempt of developing an ANN was by McCulloch and Pittman in 1943 who published a simple neural network model using electrical circuits to describe how neurons in the brain work [43]. Since then, ANNs have been used successfully as general powerful function approximators or model generators for solving various technical problems. The reliability of the UAV navigation information when the environment characteristics change was improved by Guan and Cai [44] using an ANN based on Radial Basis Functions (RBF) with Kalman filter and particle filter. The filters used for data fusion and the RBF neural network were used to estimate the error of the particle filter. When the data are available, the neural network performs the training mode, and when the data flow is interrupted or unreliable, the system uses the trained model. The accuracy of the position in two dimensions

(x and y) has been significantly increased. The AIS paradigm augmented with artificial neural networks was used by Perhinschi and colleagues [45] for developing and testing through simulation aircraft sub-system failure detection and identification schemes. The AIS paradigm included the neural estimates of the angular accelerations defined as features affected by abnormal conditions.

2.2.4. Fuzzy Logic

Fuzzy logic is an alternative generalized logic that uses continuous truth values between 0 and 1 instead of only using the binary extremes. The fuzzy logic was first introduced by Lotfi Zadeh, a professor at the University of California at Berkley. In 1965, Zadeh published his first paper on fuzzy logic entitled "Fuzzy Sets,'' which was the beginning of numerous applications of the fuzzy logic concept [46]. In 1973,Zadeh published another paper on the analysis of complex systems and decision processes and, in 1979, proposed the extensions of the possibility theory to the fuzzy information granules [47]. In 1981, Zadeh published another paper on possibility theory and soft data analysis [48].

Typically, the inputs of the fuzzy logic-based system are converted into outputs in three main steps: fuzzification, decision making, and defuzzification. Trajectory tracking for an autonomous UAV using fuzzy logic has been investigated by Perhinschi [49]. Sabo and Kelly developed a two-dimensional motion planning approach for a UAV using fuzzy logic to command the changes in heading angle and the speed [50]. The information about the target location and obstacles was sent to the fuzzy inference system in real time within the sensing range of the sensors. Sun *et al.* [51] examined path tracking and obstacle avoidance using a fuzzy logic approach. Moving and immobile obstacles were considered along with the preplanned path at each instant. Wilburn *et al.* demonstrated that a fuzzy logic-based scheme for UAV navigation possesses better capabilities compared to a potential field controller [52].

2.2.5. Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a computational stochastic optimizing technique that attempts to iteratively improve a candidate's solution with respect to a given measure of quality [53]. PSO was introduced in 1995 by Kennedy and Eberhart [54] as an optimization method to solve nonlinear problems. The PSO was inspired by the social behavior of certain bird and fish species that can coordinate the movements of many individuals quickly and accurately. PSO involves seeking the best solution in the search space. Each solution is called a particle and has a cost value that is minimized by the evaluation function. Ahmad Zadeh and Ghanavati proposed an approach for using PSO in mobile robot navigation in a dynamic environment [55]. The proper path to the goal position consisted of several points that were selected individually using the PSO optimization technique. Sensors were used to detect moving and fixed obstacles with a limited surrounding radius to successfully minimize the traveling time and distance, while avoiding obstacles.

2.2.6. The Artificial Immune System

Artificial Intelligence (AI) has proven its feasibility in many fields. Artificial intelligence techniques are inspired by various biological systems such as the immune system in superior vertebrates, which provided the ideas for the formulation of the AIS paradigm. Some of the most popular immunity concepts include the negative and positive selection algorithms, cloning and clonal selection, cellular memory, immune network theory, and danger theory. The general AIS methodology for system abnormal condition detection and identification was outlined by Dasgupta [56] including the highly robust ability of the organisms to detect, identify, and eliminate invading pathogens while overlooking its own cells.

The use of immune systems was primarily targeted at the detection and identification of the systems of abnormal and failure conditions. An AIS-based framework for aircraft abnormal condition detection, identification, evaluation, and accommodation was formulated by WVU researchers [57, 58]. The proposed methodology has the capability of providing an integrated and comprehensive solution to the problem of aircraft monitoring and control under normal and abnormal operational conditions [59]. Specific immunity-inspired approaches for abnormal condition detection, identification, evaluation, and accommodation have been developed. In particular, mimicking the capability of the immune system to memorize antigen/antibody

correlation was investigated with promising results for aircraft control purposes [60]. The immunity-based monitoring approach was successfully tested with actual UAV flight data [61] and hardware-in-the-loop simulation [62, 63]. Navigation and obstacle avoidance for a mobile ground robot was approached by Ozcelik and Sukumaran [64] using AIS; the obstacle's position was assimilated to antigens and the required vehicle heading for adequate obstacle avoidance to the antibodies.

2.3. Global Navigation Satellite System

The United Nations defined the GNSS at the Third United Nations Conference on the Exploration and Peaceful Uses of Outer Space in 1998 as follows: "The Global Navigation Satellite System (GNSS) is a space-based radio positioning system that includes one or more satellite constellations, augmented to support the intended operation. The GNSS provides 24-hour three-dimensional position, velocity, and time information to suitably equipped users anywhere on or near the surface of the Earth (and sometimes off Earth)" [65].

The US military conceived the Navy Navigation Satellite System (NNSS) concept in the late 1950s and developed it in 1960s mainly for determining the time and coordinates of vessels at sea and military application on land. Eventually, the NNSS became authorized for civilian uses such as navigation and surveying. The US military later overcame the shortcomings of the early navigation systems and developed the Navigation System with Timing and Ranging (NAVSTAR) or GPS. The Russian military developed its counterpart to GPS, the Global Navigation Satellite System (GLONASS) while European countries also contributed to the GNSS with Galileo. Finally, the Chinese GNSS called BeiDou (Compass) is the first-generation regional system [65]. The principle of GNSS is based on a trigonometry solution of a geometrical problem to determine positions using Earth stations as reference points.

The receiver is located at the intersection of four spheres; each centered at known coordinates of the satellite that broadcast information to the receiver [12]. The pseudorange of each satellite defines the surface of a sphere with its center at the satellite position [65]. The radius of each sphere represents the distance between the satellite and the receiver or the pseudorange. Three satellites are needed to measure the latitude, longitude, and height which can be determined using three range equations; however, there is an offset between the receiver's clock and the true system time usually caused by the inexpensive crystal clock used in the

receiver. Therefore, a fourth satellite is needed to solve the four unknowns, namely three components of position plus the clock bias [65]. An integrated GNSS and vision sensor suite approach was used by Roberts [66] for aircraft collision and obstacle avoidance and navigation. The GNSS contribution was used to provide primary aircraft navigation, and the vision sensor was to provide obstacle and aircraft collision avoidance. Cho *et al.* [67] assumed that the inertial sensors such as gyros and accelerometers were disabled in a UAV and single-antenna GPS receiver based attitude determination system was used for fully automatic control of taxiing, landing, and take off on the runway for UAV. The single-antenna GPS receiver can be used as a primary sensor for a backup or a low-cost control system of UAVs.

2.4. GNSS Vulnerability

The distances between satellites and receiver or pseudoranges are calculated using the Time of Arrival (TOA) concept. TOA refers to the time it takes the electromagnetic wave signal of the satellite to propagate and reach the receiver within the shortest straight path. In other words, TOA measures the differences between the satellite and receiver clock times when the signal arrives at the receiver. The measured TOAs for each satellite multiplied by the speed of light to calculate the pseudoranges which are used in trigonometrical equations to calculate the user position [12]. Ideally, the satellite signal propagates in a vacuumed surrounding at a speed of light in an obstacle-free path without interferences, hardware faults and failures, and the clocks of the satellites and receiver are precisely synchronized. Any alteration to the ideal scenario causes errors and contributes to inaccurate user position [68]. In standard stand-alone operations of GNSS, TOAs provide a three-dimensional position of approximately 5-10 meters accuracy depending on user equipment infrastructure, error sources, and configuration for tracking.

2.4.1. Uncertainty Sources in GNSS

The GNSS signals are transmitted through different layers of the atmosphere such as the ionosphere and troposphere. These transmission media cause deviations of the traveling signal from the shortest path, increasing the travel time and causing position calculation errors [69]. The GNSS is affected by the ionospheric storms over the Equator; therefore, augmentation systems have been developed to make corrections to the message transmitted to user receivers [70]. For military uses, GPS signals are transmitted at two different frequencies to mitigate the ionospheric effect. Unfortunately, this service is not available for commercial non-authorized

users. However, several strategies such as using mathematical models and using additional information provided by ground, and space-based augmentations can be used to mitigate the ionospheric effect [71]. The troposphere ranges from the Earth surface to 17–20 km in altitude, whereas the ionosphere lies between 75 and 500 km. The difference between the expected and actual orbital position of a GNSS satellite determines the ephemeris errors, which create about a few meters of error in computing position. The geometric arrangement of satellites, as they are "visible" to the receiver, affects the measurement through Dilution of Precision (DOP), which is a metric of geometric diversity of the satellites and relates parameters of the user position and time bias errors to those of the pseudorange errors. The DOP contributes to navigation accuracy and proper measurements can be obtained if adequate user/satellite geometry exists [12]. Receivers are designed to use signals from available satellites in a manner that minimizes this effect. Errors due to DOP are caused by poor positioning of satellites; thus, proper distribution of satellites produces better navigation [12]. If signals arrive over two or more paths due to multiple reflections, interferences in the receiver occur and affect the signal quality. Another cause of interfering is the weak signal from the satellite.

2.4.2. Radio Frequency Interference in GNSS

The GNSS satellites broadcast signals in two or more frequencies in L band, and typically these signals are received at low power on the Earth's surface [72]. These signals can be disrupted by other Radio Frequency (RF) signals that appear in the bands. The interfering signals come intentionally or unintentionally from transmitting devices, causing poor GNSS receiver performance [6]; therefore, the receiver needs to detect these interferences to preserve adequate performance. The GNSS receivers can be interfered unintentionally by malfunctioning radio devices when they operate at the subharmonics of GNSS carrier frequencies [6]. The intentional interference is caused by devices intended to disrupt the normal GNSS operation and is referred to as jamming, spoofing, and meaconing [65]. Jamming involves transmitting a high-power signal close to the GNSS signal by jammer devices to mask it out and disable it from properly entering the receiver. Spoofing refers to a counterfeit signal that tricks the GNSS receivers to produce faulty information [12]. Spoofing is more challenging than jamming because the faked signal attempts to mimic a valid one and typically cannot be detected until a severe problem occurs, whereas the loss of signal caused by jamming is more apparent [73].

Meaconing involves attacking the GNSS integrity by delaying and rebroadcasting (recording and playback) the block of RF spectrum that contains the GNSS signals [6]. Gross and Humphreys [74] used a multiple-correlation tap, a maximum-likelihood multipath estimator to detect and classify GNSS interference. The proposed method was used for warning against the occurrence of GNSS spoofing, jamming or multipath.

2.5. Inertial Measurement Unit (IMU)

Inertial Measurement Units (IMUs) consist of a combination of specific forces and angular rates sensors, usually three accelerometers that produce three-dimensional measurements of the specific force and three gyroscopes to measure the angular rates. The IMU also has a processor, a set of calibration parameters, a temperature sensor, and the associated power supply [30]. The main errors of the IMU units are biases, scale factor and cross-coupling, and random noise. The biases for accelerometers and gyros are denoted by the vector $\vec{b}_a$= [$b_{a,x}$, $b_{a,y}$, $b_{a,z}$], and $\vec{b}_g$= [$b_{g,x}$, $b_{g,y}$, $b_{g,z}$]. There are two types of biases; static known as fixed, and turn-on or repeatability and dynamic bias known as instability bias or in-run bias variation, such that the total bias: $\vec{b}_a = \vec{b}_{as} + \vec{b}_{ad}$, and $\vec{b}_g = \vec{b}_{\mathrm{gs}} + \vec{b}_{\mathrm{g}d}$. The dynamic bias is approximately 10% of the magnitude of the static bias. The unit of measuring the accelerometer bias is milli-g (mg) or micro-g (μg) where 1g is 9.80665 ms^{-2}. For gyroscope bias, the unit is degree per hour ($^{o}hr^{-1}$) or rad per second (rad s^{-1}). Scale factor errors related to the unit conversion by the IMU and can be denoted as; $\vec{s}_a = [s_{a,x}, s_{a,y}, s_{a,z}]$ for accelerometers and $\vec{s}_{\mathrm{g}} = [s_{\mathrm{g},x}, s_{\mathrm{g},y}, s_{\mathrm{g},z}]$ for gyros. The cross-coupling error is caused by the misalignment of the IMU's sensors axis and denoted as $m_{a,\alpha\beta}$ for accelerometer and $m_{\mathrm{g},\alpha\beta}$ for gyros, where β and α represent the axis of misalignment. The scale factor and the misalignment errors can be represented as the following matrices:

$$\mathrm{M_a}= \begin{bmatrix} s_{a,x} & m_{a,xy} & m_{a,xz} \\ m_{a,yx} & s_{ay} & m_{a,yz} \\ m_{a,zx} & m_{a,zy} & s_{a,z} \end{bmatrix},\ \mathrm{M_g}= \begin{bmatrix} s_{\mathrm{g},x} & m_{\mathrm{g},xy} & m_{\mathrm{g,xz}} \\ m_{\mathrm{g},yx} & s_{\mathrm{g},y} & m_{\mathrm{g,yz}} \\ m_{\mathrm{g,zx}} & m_{\mathrm{g,zy}} & s_{\mathrm{g},z} \end{bmatrix} \qquad (1)$$

The random noise denoted as $\vec{w}_{\mathrm{a}}$ = [$w_{\mathrm{a,x}}$, $w_{\mathrm{a,y}}$, $w_{\mathrm{a,z}}$] for accelerometers and $\vec{w}_{\mathrm{g}}$ = ($w_{\mathrm{g,x}}$, $w_{\mathrm{g,y}}$, $w_{\mathrm{g,z}}$) for gyros with a unit of random noise is μg/$\sqrt{Hz}$ for accelerometer and $^{o}/\sqrt{Hz}$ for gyro. The equations of the accelerometer and gyro errors model are:

$$\text{Accelerometer errors} = \vec{a}_m - \vec{a} = (\vec{b}_a + (I_3 + M_a)\vec{a} + \vec{w}_a) - \vec{a} \quad (2)$$

$$\text{Gyro errors} = \vec{\omega}_m - \vec{\omega} = (\vec{b}_g + (I_3 + M_g)\vec{\omega} + G_g\vec{a} + \vec{w}_g) - \vec{\omega} \quad (3)$$

$\vec{a}_m$ and $\vec{\omega}_m$ are the IMU's outputs, the measured acceleration and angular rate vectors, while $\vec{a}$ and $\vec{\omega}$ are the true counterparts. I_3 is the identity matrix and G*g* is a 3x3 matrix representing the gyro sensitivity to accelerations along all three axes [30]. Hardy [75] estimated the relative pose between two UAVs operating in GNSS-denied environments by fusing multiple on-board sensors using an unscented Kalman filter (UKF). The sensitivity of the navigation algorithm was investigated using Monte Carlo simulation. Du *et al*. [76] installed a Micro-Electro-Mechanical System (MEMS) grade IMU on a rotation platform referred to as a rotary inertial navigation system. This system can mitigate the MEMS navigation errors when external aiding information is not available. The time was increased and calibration applied to remove the IMU rotation induced errors.

Pinpoint landing system for Mars entry requires a high accuracy navigation system. The lack of accuracy for this navigation system was discussed by Lou [77]. A Schmidt-Kalman filter was formulated to mitigate the effects of systematic bias errors. The cross-correlation between the states and the measurement bias were considered, leading to realistic covariance estimate. Matrix inversion operation was avoided by implementing the algorithm of the upper matrix and diagonal matrix (UD) decomposition in the Schmidt-Kalman filter, and the numerical stability of the filtering was insured. Monte Carlo simulation was used to show the quality of the Schmidt-Kalman filter performance with UD decomposition. The Schmidt-Kalman filter has significantly improved state accuracy.

Rhudy et al. [78] presented a vision-aided inertial navigation technique that relied on inertial sensors and wide-field optical flow information. An unscented information filter was used to estimate an aircraft ground velocity and attitude states. The states were evaluated in relation to two sets of UAV experimental flight data. Within each formulation, an additional state was considered to recover the image distance, which can be measured using a laser range finder. Two formulations were assumed; full state formulation including velocity and attitude, and a simplified formulation neglecting the lateral and vertical velocity. Both formulations showed effective results in regulating the inertial navigation system drift[78].

3. General Formulation of the AIS-based Framework

3.1. The AIS Paradigm

The AIS paradigm [23] mimics the biological immune system mechanisms capable of detecting and eliminating pathogens without hurting the self-cells [24]. Within the human body, the biological immune system protects the body from dangerous pathogens. Immunity mechanisms allow for the localization and identification of the affected area, facilitating the concentration and effectiveness of defensive responses. The biological immune system is a naturally occurring adaptive control system that consists of specialized organs, cells, and chemical compounds whose ultimate objective is protecting the organism (the self) against harmful invading agents such as bacteria, viruses, and parasites (the non-self). The latter are generically referred to as antigens, while the specialized cells directly active in destruction are referred to as antibodies. Specialized immunity cells are generated based on the positive and negative selection, which attempt to ensure compatibility with the self and affinity to non-self, respectively. The immune system must first detect the presence of antigens by discriminating between self and non-self-based chemical markers that are present on the antigen's surface but are not present on other cells. The number and virulence of the antigens can be assessed; thereby, facilitating the determination of the danger of the invasion. This evaluation of the situation is used to govern the humoral feedback mechanism which is responsible for controlling the generation of antibodies at levels required for destroying the antigen while using resources sparingly. The immune system comprises antibodies and lymphocytes. The two types of lymphocytes in the immune system are B-cells and T-cells. B-cells are produced in the bone marrow and used to recognize and eliminate antigens by generating antibodies. The thymus produces two types of T-cells, suppressing T_s-cells and assistant or helper T_h-cells. The T-cells are vital in regulating the production of B-cells and antibodies. If many antigens are detected, more T_h-cells and fewer T_s-cells are produced, resulting in more B-cell production. As the number of antigens present in the body decreases, the number of suppressing T_s-cells increases, while T_h-cells are reduced, ultimately leading to fewer B-cells and antibodies[24]. Eventually, this process equalizes, and the immune response is complete. An overview of this process is presented in Figure 1.

Some of the immunity mechanisms have inspired the establishment of a framework by improving the capability for compensating errors of position and velocity estimation based on inertial measurement when GPS information or alternative sources are not available. Artificial memory cells are built as tandem collections of values of the system states and estimation errors or corrections. System state values are assimilated to the antigens because they are directly correlated to the estimation corrections, while the latter are assimilated to the antibodies. A positive selection type of mechanism must be used to select the proper memory cell by matching its state values to the current measured state. Once this is accomplished, the needed correction for the estimation scheme can be extracted from the artificial memory cell. This process is equivalent to the detection of antigen by specialized immune cells and the use of memory B-cells for accelerated reaction through antibody generation. The AIS exhibits highly robust and adaptive classification and information structuring capabilities, as well as memory and information fusion potential.

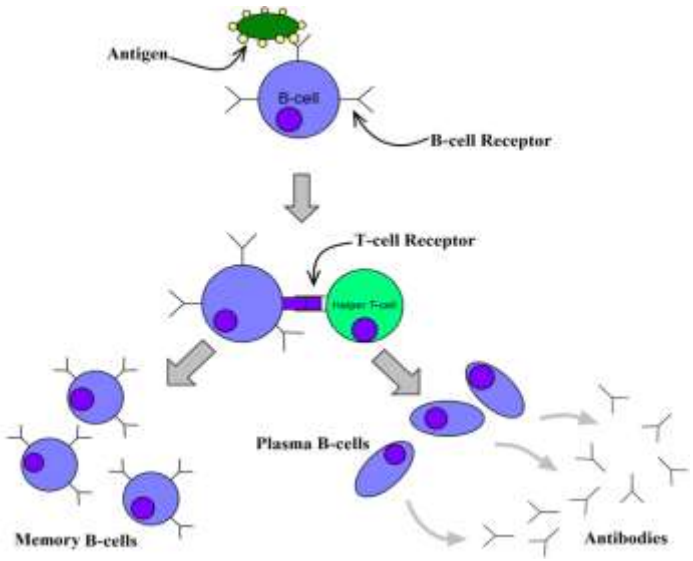

Figure 1. Immunity cell interactions.

3.2. Problem Formulation

An autonomous UAV is assumed to follow a commanded trajectory with disabled GPS feedback information. The measured states such as accelerations and angular rates are measured

on board by adequately functioning sensors to be used in the UAV velocity and position estimation scheme. Through integration, the estimation errors accumulate over extended periods. The UAV would not achieve an adequate tracking performance because of such accumulative errors. It was hypothesized that an improvement of the trajectory tracking performance could be achieved by immediate estimated error corrections that depend on the short-term modification of vehicle dynamic state. The estimated corrections are assumed to be provided by the AIS. Within the AIS paradigm, the antigens represent the short-term state modifications, while antibodies represent the needed corrections. Therefore, the AIS construction consists of a collection of memory cells that associate to each configuration of dynamic system change, a set of instantaneous estimation corrections. The variables that can be measured on-board must define the dynamic system state. When the measured velocity and position are available, the corrections can be determined in flight as an estimation error. For comprehensive usage of the approach, the AIS flight test construction must ideally cover all possible vehicle state configurations.

3.3. Definitions and Notations

3.3.1. The Aircraft (or UAV) Reference Frame and Coordinate System

The aircraft is assumed to be associated with the reference frame RF_B and has a rigid body and constant mass. The "body axes" are the aircraft Coordinate System (*CS*) and denoted as CS_B or $OX_B\ Y_B\ Z_B$ (see Figure 2). The origin O of the body axes is at the center of mass of the aircraft. The longitudinal axis X_B is along the fuselage with the positive direction forward in the aircraft plane of symmetry with a direction at the discretion of the designer. Y_B is the lateral axis positive to the right of the pilot and the vertical axis Z_B is positive downward, as dictated by the right-hand rule.

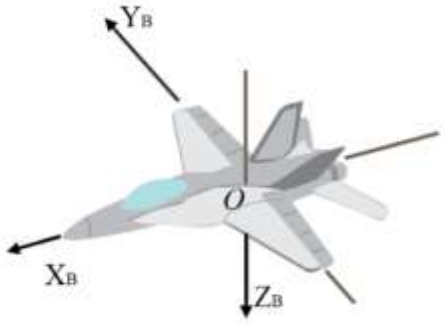

Figure 2. The aircraft reference frame.

3.3.2. Earth Reference Frame and Coordinate System

The CS relative to the Earth is denoted as CS_E or $E\ X_E\ Y_E\ Z_E$. The Earth is assumed to be flat and inertial. E is the origin and established by the user on the interactive map of WVU UAS simulation environment. The initial location of the aircraft center of mass coincides with the origin E. X_E is the longitudinal earth axis and selected to point up (towards North) with respect to the simulation environment map. Y_E is the lateral axis and is in a positive direction to the right (Eastwards). Z_E is positive into the plan of the map as presented in Figure 3. The components of the velocity vector $\vec{v}$ with respect to CS_E (or $E\ X_E\ Y_E\ Z_E$) will be denoted as $[\vec{v}]_E = [v_x \quad v_y \quad v_z]_E^T$.

Figure 3. Earth reference frame.

3.3.3. Euler Angles

Three Euler angles define the relative orientation of two CSs. Three rotations must be applied to obtain Euler angles successively along one axis at a time applied to one CS such that it eventually overlaps the second. The typical order of rotation starts with the vertical axis followed by the lateral and longitudinal axes [79]. For an aircraft, Euler angles represent the orientation of the fixed body axes with respect to a CS fixed in relationto the Earth. Euler angles are also referred to as aircraft attitude angles or angular position and denoted as roll, pitch, and yaw attitude angles (φ, θ, ψ, respectively). With the most commonly used conventions, the roll attitude angle is positive if the aircraft is tilted to the right of the pilot, the pitch attitude angle is

positive if the aircraft is tilted nose-up, and the yaw attitude angle is positive if the nose of the aircraft points to the right of the pilot (see Figure 4).

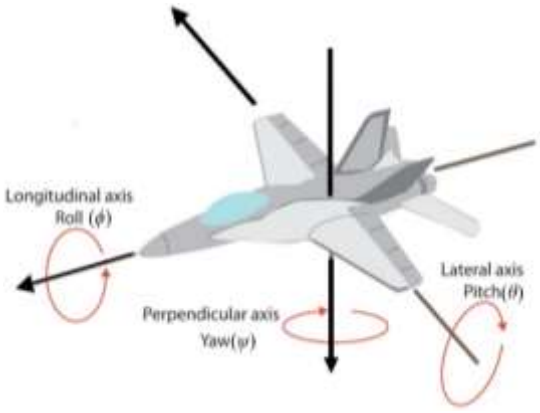

Figure 4. Roll, pitch, and yaw rotation axes.

3.3.4. Transformation Matrix

When the coordinates for one CS are known, then the components or coordinates of a vector in relation to a different CS can be obtained using a transformation matrix. The transformation matrix from CS_E to CS_B is denoted as L_{BE}. The transformation matrices are orthonormal or $L_{BE}^{-1} = L_{BE}^{T}$. The elements of the transformation matrices are trigonometric functions of the Euler angles between the two CS.

$$[\vec{v}]_B = L_{BE}[\vec{v}]_E = L_{EB}^{-1}[\vec{v}]_E \tag{4}$$

The transformation matrix from body axes components to Earth axes components is:

$$L_{EB} = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix} \tag{5}$$

3.3.5. Vector Derivative

Both variations in the magnitude and orientation of the vector $\vec{v}$ must be taken into consideration; therefore, the derivative of a vector is defined in relation to the reference frame RF_E and denoted as:$\frac{^E d\vec{v}}{dt}$ [80].

The relationship between the derivatives of the same vector with respect to two different RFs (RF_E and RF_B) is represented as:

$$\frac{^E d\vec{v}}{dt} = \frac{^B d\vec{v}}{dt} + {}^E\vec{\omega}^B \times \vec{v} \tag{6}$$

where ${}^E\vec{\omega}^B$ is the rotation vector of RF_B with respect to RF_E.

3.3.6. Position Vector

$\vec{p}^{EO}$ is the denotation of the position vector of the aircraft center of mass *O* with respect to a reference point on Earth, *E*. The UAV trajectory is defined by the position vector which is directed towards *O* and its origin is at point *E*. In other words, its components in RF_E are used to define the trajectory as follows:

$$[\vec{p}^{EO}]_E = \begin{bmatrix} x_E \\ y_E \\ z_E \end{bmatrix}_E \tag{7}$$

3.3.7. Velocity Vector

${}^E\vec{v}^O$is the denotation of the translational velocity of the aircraft center of mass *O* with respect to Earth reference frame RF_E and can be represented as follows:

$${}^E\vec{v}^O = \frac{^E d\vec{p}^{EO}}{dt} \tag{8}$$

where $\frac{^E d}{dt}$ is the derivation operator with respect to RF_E and *E* can be any fixed point in RF_E. The components of ${}^E\vec{v}^O$ with respect to RF_E and RF_B are denoted as follows:

$$[{}^E\vec{v}^O]_E = \begin{bmatrix} \dot{x}_E \\ \dot{y}_E \\ \dot{z}_E \end{bmatrix}_E = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}_E, [{}^E\vec{v}^O]_B = \begin{bmatrix} u \\ v \\ w \end{bmatrix}_B, \text{ and } [{}^E\vec{v}^O]_E = L_{EB}[{}^E\vec{v}^O]_B \tag{9}$$

where: u, v, and w are the velocity components in x_B, y_B, and z_B directions receptively.

3.3.8. Acceleration Vector

${}^{E}\vec{a}^{O}$ is the denotation of the translational acceleration of the aircraft center of mass O with respect to Earth reference frame RF_E, as follows:

$$ {}^{E}\vec{a}^{O} = \frac{{}^{E}d\,{}^{E}\vec{v}^{O}}{dt} = \frac{{}^{E}d^{2}\vec{p}^{EO}}{dt^{2}} \tag{10} $$

Also:

$$ {}^{E}\vec{a}^{O} = \frac{{}^{E}d\,{}^{E}\vec{v}^{O}}{dt} = \frac{{}^{B}d\,{}^{E}\vec{v}^{O}}{dt} + {}^{E}\vec{\omega}^{B} \times {}^{E}\vec{v}^{O} \tag{11} $$

$$ [{}^{E}\vec{a}^{O}]_{E} = L_{EB}\left\{ \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix}_{B} + \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix}_{B} \begin{bmatrix} u \\ v \\ w \end{bmatrix}_{B} \right\} \tag{12} $$

where:

$$ [{}^{E}\vec{\omega}^{B}]_{B} = \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{B} \tag{13} $$

and :

$$ [{}^{E}\vec{a}^{O}]_{E} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}_{E} \tag{14} $$

3.3.9. Angular Position

The Euler angles $[\varphi \quad \theta \quad \psi]^{T}$ define the aircraft angular position or attitude. These are the angles that define the relative orientation of RF_B or the aircraft rigid body with respect to RF_E.

3.3.10. Angular Velocity Vector

The rotation of the aircraft as a rigid body (or equivalently RF_B) with respect to the Earth's rigid body (or equivalently RF_E) can be defined by the aircraft angular velocity vector

denoted as ${}^{E}_{1}\vec{\omega}^{B}_{1}$. The components of the angular velocity vector are represented by equation (15) are the result of three rotations about three non-perpendicular axes, consistent with the definition of Euler angles:

$$ {}^{E}\vec{\omega}^{B} = \dot{\vec{\psi}} + \dot{\vec{\theta}} + \dot{\vec{\varphi}} \tag{15}$$

Therefore:

$$ \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{B} = \begin{bmatrix} 1 & 0 & -sin(\theta) \\ 0 & cos(\varphi) & sin(\varphi)cos(\theta) \\ 0 & -sin(\varphi) & cos(\varphi)cos(\theta) \end{bmatrix} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{16}$$

and:

$$ \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & sin(\varphi)tan(\theta) & cos(\varphi)tan(\theta) \\ 0 & cos(\varphi) & -sin(\varphi) \\ 0 & sin(\varphi)/cos(\theta) & cos(\varphi)/cos(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{B} \tag{17}$$

3.3.11. Angular Acceleration Vector

${}^{E}\vec{\omega}^{B}$ vector represents the aircraft angular velocity and is invariant with respect to the derivation RF because the cross product of a vector with itself is 0.

$$ {}^{E}\vec{\omega}^{B} = \frac{{}^{E}d\,{}^{E}\vec{\omega}^{B}}{dt} = \frac{{}^{B}d\,{}^{E}\vec{\omega}^{B}}{dt} \tag{18}$$

3.3.12. Actual Values

The values of a variable that are reached by the real system are referred to as the actual values. Precisely, the actual values are not accessible because the only available access is to the measurements. To make this distinction, the actual values are referred to as the measurand values and are denoted with symbols without additional subscripts. For example, the actual position of aircraft will be designated by $\vec{p}^{EO}$. In the simulation, actual values are obtained from the mathematical model is rather of the system. To be more rigorous, the mathematical model is rather producing actual values plus modeling errors.

3.3.13. Measured Values

The sensor output values of a variable are represented as the measured values and denoted with symbols with subscript m. They are affected by measurement errors $\vec{e}_m$. For example, the measured position of the aircraft (e.g., provided by GPS) will be designated by $\vec{p}_m^{EO}$ and:

$$\vec{p}_m^{EO} = \vec{p}^{EO} + \vec{e}_m \tag{1}$$

3.3.14. Estimated Values

The estimation algorithms use limited sets of measurements, and the output values of a variable are represented as the estimated values and are denoted with symbols using subscript e. Estimation algorithms are affected by estimation errors $\vec{e}_e$ that are assumed to be the sum of estimation algorithm errors and measurement errors. For example, the estimated position of the aircraft will be designated by $\vec{p}_e^{EO}$ and:

$$\vec{p}_e^{EO} = \vec{p}^{EO} + \vec{e}_e \tag{2}$$

3.3.15. AIS-corrected Estimated Values

AIS-corrected estimations are the values obtained from the estimation scheme plus the corrections extracted from the AIS. For example:

$$\vec{p}_{ce}^{EO} = \vec{p}_e^{EO} + \vec{p}_{AIS}^{EO} \tag{3}$$

3.3.16. Artificial Memory Cells

The Artificial Memory Cells (AMC) represent the AIS paradigm memory. The AMC mimic the T- and B-cells in the biological immune system, which stores information about antigen characteristics and the required antibodies for future accelerated and stronger immune response. AMCs are the building blocks of the AIS and consist of two parts: One including information on the characteristics of the system dynamic state referred to as “antigens” and one including the corresponding corrections for the estimation output, referred to as “antibodies.”

3.4. AIS-based Framework Architecture

The AIS-based framework relies primarily on two main processes:

i. Building the AIS which is performed "off-line" with data acquired during normal operation of the system.

ii. Extracting position and velocity estimation corrections from AIS and applying them, which is performed "on-line," when GPS is not working.

The AIS is envisioned as an information depository that relates states of aircraft with corrections needed for position and velocity estimations. Therefore, two sets of AIS features are needed:

i. The UAV features, including variables that define the dynamic state of the vehicle.

ii. The correction features, including the variables that represent corrections needed for the position and velocity estimations.

The estimation scheme of the position and velocity is assumed to operate adequately with information from on-board sensors, excluding GPS or alternative sources. Estimation scheme output is necessary both for the acquisition of data for AIS generation and during operation without GPS. The same level of performance of the estimation scheme is assumed in both situations.

3.4.1. The AIS Generation

The generation of the AIS consists of collecting and processing data under nominal conditions and structuring them as a set of artificial memory cells. Figure 5 presents the AIS building process. The main steps in this process are:

i. Definition of UAV and correction features.
ii. Design of tests for collecting data covering all possible dynamic configurations.
iii. Execution of tests and data collection.
iv. Processing data for normalization and duplicate elimination.
v. AIS structuring as a table with two distinct areas for antibodies and antigens.

The following variables must be recorded for the commanded trajectory:

i. Sensor measurements of UAV features.

ii. Estimated position and velocity produced by the on-board estimation scheme.

iii. Measured position and velocity from GPS or other available sources.

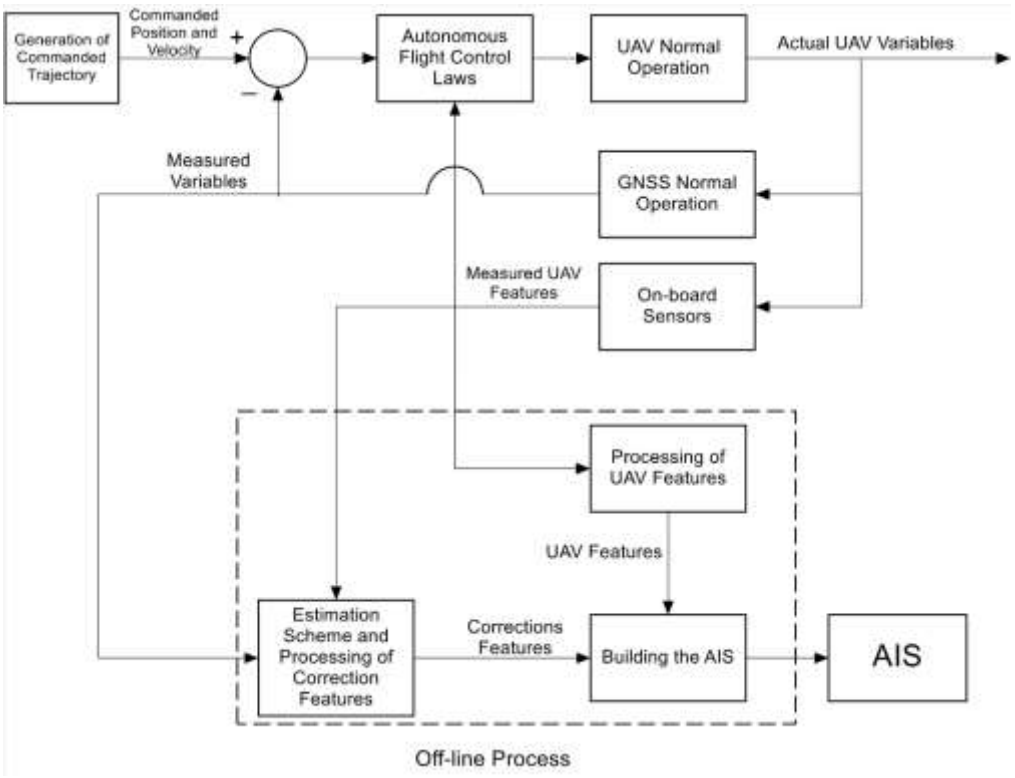

Figure 5. Building the AIS (off-line).

Two alternatives for structuring the AIS-based estimation correction process have been identified. One approach involves the correction of the output of the position and velocity estimation scheme (estimated vehicle position and velocity) and the other, the correction of the input of the estimation scheme (measured acceleration). The first approach will be referred to as the “output” approach and the second, as the “input” approach. The two alternative solutions are illustrated in Figures 6 and 7 respectively.

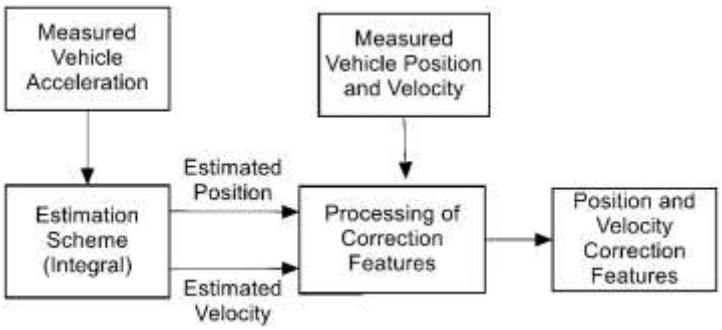

Figure 6. Estimation scheme and processing corrections for the "output" approach.

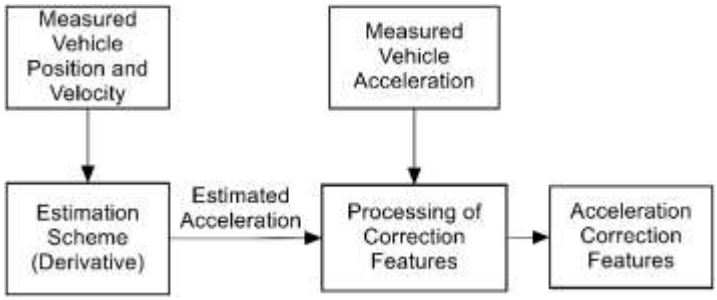

Figure 7. Estimation scheme and processing corrections for the "input" approach.

The AIS "memory cells" sets:

The AIS consists of a set of "memory cells" that mimic the functionality of memory T- and B-cells that can collect and memorize information about intruding antigens as well as information on the corresponding antibodies that can fight the antigens. Such an artificial "memory cell" may be built as a simultaneous sample of UAV features and correction features. Alternatively, a memory cell may consist of a UAV feature cluster and associated values of the correction features.

The UAV features:

The UAV features define the state of the system and are expected to be correlated to the needed corrections. In this research, the candidate set of UAV features that were used for the example implementation are the increments of angular attitude rates (p, q, r) and the acceleration components(a_x, a_y, a_z). The UAV feature values represent the antigen within the AIS paradigm. Note that UAV features can be extended to include a possibly large number of variables.

The correction features:

Within the AIS paradigm, the correction feature values represent the response antibody. The idea is that the AIS has a corrective countermeasure to each possible situation captured by UAV features. The correction features may include different variables depending on the structure of the estimation scheme and the actual correction mechanism; however, the ultimate purpose is to obtain corrected estimations of vehicle position and velocity.

Let the AIS features be denoted as $\varphi_i,\ i = 1,2,\dots,N$. The set of all AIS features φ consists of two disjunct subsets, the system or UAV featuresφ_s (representing the antigens) and the correction features φ_c (representing the antibodies):

$$\varphi = \varphi_s \cup \varphi_c \tag{22}$$

Alternatively:

$$\{\varphi_i \mid i = 1,2,\dots,N\} = \{\varphi_{si} \mid i = 1,2,\dots,N_s\} \cup \{\varphi_{ci} \mid i = 1,2,\dots,N_c\} \tag{23}$$

where $N = N_s + N_c$. For building memory cells, samples Σ must first be recorded consisting of two concatenated strings of values taken by the two types of features, respectively, at each sampling instant k ($k = 1,2,\dots,N_\Sigma$). Therefore:

$$\Sigma^{(k)} = \left[\varphi_{s1}^{(k)} \varphi_{s2}^{(k)} \quad \dots \quad \varphi_{sN_s}^{(k)} \ \vdots\ \varphi_{c1}^{(k)} \varphi_{c2}^{(k)} \quad \dots \quad \varphi_{cN_c}^{(k)}\right] \tag{4}$$

A memory cell $M_j, j = 1,2,\dots,N_M$ consists of such a sample; therefore:

$$M_j = \Sigma^{(k)} \tag{5}$$

Note that typically $N_M \ll N_\Sigma$, if duplicate data points are removed.

An alternative clustering approach could be used to define the structure of the AIS [57, 58]. The samples of the vehicle features would have to be clustered and associated with a set of correction values. If hyper-spherical clusters are considered, the AMC would consist of a set of clusters (*N*-dimensional center and scalar radius) and the associated estimation corrections. The

approach would reduce the memory necessary to host the AIS and the time required for the matching algorithm. The clustering approach would also facilitate integration with other processes for which the AIS paradigm represents a promising solution, such as abnormal condition detection, identification, and evaluation [59].

3.4.2. AIS Compensation

Matching algorithm:

The biological immune system can differentiate between self and non-self cells [81]. Similarly, for the AIS, a matching algorithm is used to identify the memory cell to be used in the correction algorithm by matching the current measured UAV features to the antigen. Assuming the sampling approach, the affinity between the normalized values of UAV feature from antigens of the AMC (a) and the normalized values of UAV feature from IMU (b) is related to their relative distance that can be estimated via any distance measure between two vectors.

Euclidean Affinity:

$$D = \sqrt{\sum_{i=1}^{L} (a_i - b_i)^2} \tag{26}$$

Manhattan Affinity:

$$D = \sum_{i=1}^{L} |a_i - b_i| \tag{27}$$

Hamming Affinity:

$$D = \sum_{i=1}^{L} \delta, \text{ where} \delta = \begin{cases} 1 & if\ a_i \neq b_i \\ 0 & otherwise \end{cases} \tag{28}$$

With the clustering approach using hyper-spheres, “a” would represent the center of the cluster and distance must be compared to the radius to establish if the current feature point falls inside the cluster or not.

The correction scheme:

The extracted correction or antibody is used to correct the output or the input of the estimation scheme.

The estimation scheme:

In this research, the measured acceleration from the UAV sensors was integrated to estimate the vehicle velocity and position; however, the estimation scheme needs correction to

compensate for the cumulated errors to ensure successful trajectory tracking. The AIS will provide these corrections.

The feedback loop:

The corrected estimation is then used in the feedback loop instead of the GPS output. Figure 8 implements the process of on-line AIS-based compensation. Figures 9 and 10 illustrate the correction process of output and input of the estimation schemes, respectively.

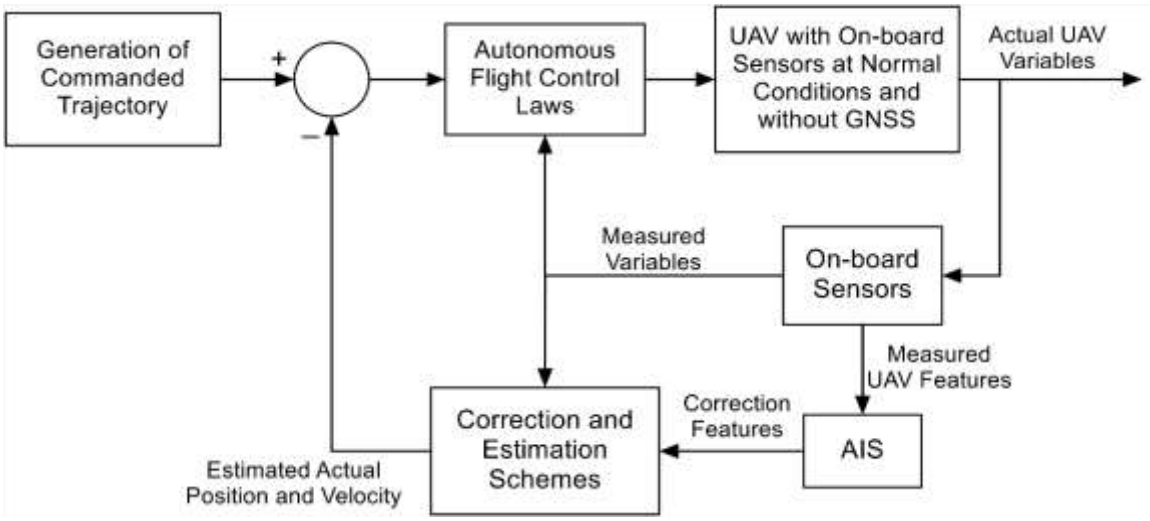

Figure 8. On-line AIS compensation.

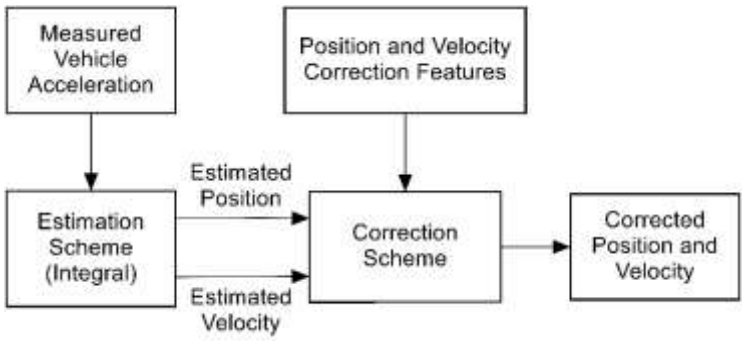

Figure 9. Correction of the output of the estimation scheme

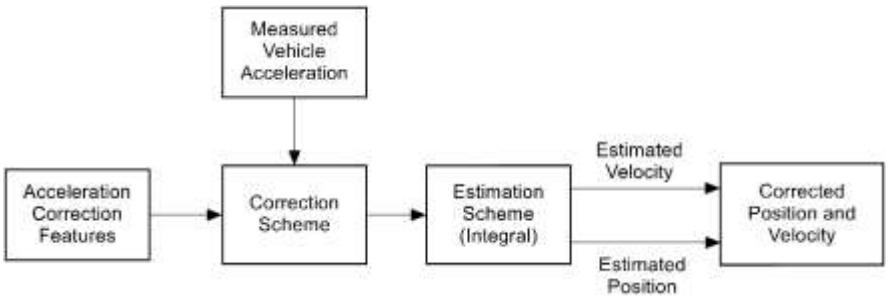

Figure 10. Correction of the input of the estimation scheme.

Let $\varphi_s^{(\propto)}$ and $\mathrm{E}^{(\propto)}$ be the set of values of vehicle features and the output of the position and velocity estimation algorithm, respectively, at the current moment $\propto$. $\varphi_s^{(\propto)}$ is referred to as the current antigen and must be matched to the closest memory cell. The memory cell M_j is first identified. If the sampling approach is used, the current antigen is matched to the stored antigens such that the distance between $\varphi_s^{(\propto)}$ and the AIS vehicle feature point $\varphi_s^{(k)}$ is minimal. If the clustering approach is used, then the cluster in which the current vehicle feature point (or antigen) resides must be determined. In both approaches, the corrections $c^{(\propto)} = \varphi_s^{(k)}$ corresponding to the identified AMC are extracted and used to obtain corrected estimations $\bar{\mathrm{E}}^{(\propto)}$ to be used in the feedback loop. In general, this corrective function can be expressed as:

$$\bar{\mathrm{E}}^{(\propto)} = \mathcal{F}\left[\mathrm{E}^{(\propto)}, c^{(\propto)}\right] \tag{29}$$

3.5. AIS Paradigm Challenges

The AIS is a novel computational artificial intelligence technique that takes inspiration from the functionality and the mechanisms of the biological immune system, without necessarily attempting to model the biological counterpart accurately. This is a general characteristic of most, if not all, biomimetic approaches. It should also be added that in many cases, the biological phenomena are not entirely understood and deciphered. This raises at least two critical issues. One is that lacking accuracy concerning the source of inspiration may impact reaching the expected desirable results. The other is that biological systems may sometime fail themselves, due to

characteristics that may be transferred to the technical implementation and produce undesirable effects.

It is hypothesized that the antibody or the needed correction depends on the antigens or the current state of the system at each instant, in another word, there should be a correlation between the antigens and the antibodies to get the required recovery for the system. Identifying the right antigens or system features that influence the system malfunctioning is a critical premise for the application of the AIS framework. Therefore, considering antigens that do not affect the malfunctioning of the system or not considering antigens that do, is expected to reduce the performance of the AIS significantly. As a consequence, careful analysis supported by test data is needed to select the proper set of antigens or system features that correlate well with the antibodies or correction features. This process must primarily rely on heuristics and experiments; however, an approach for assessing the potential of different features is proposed and investigated within this research effort.

The data used for generating the AIS must ideally cover all possible dynamic scenarios in conjunction with correction patterns and distributions. If proper coverage of the feature hyperspace is not achieved, a very significant problem may occur because the affinity or matching algorithm between the current antigen and memory cells may lead to the extraction of wrong corrections. Therefore, the generation of AIS should rely on enough data to obtain satisfactory results. It should be noted that the AIS can be continuously updated during system functioning, once additional data are available, and a simple evaluation scheme can be implemented based on the level of antigen mismatch to assess if the AIS coverage is adequate or not at any given moment.

4. The WVU UAS Simulation Environment

An advanced simulation environment has been developed at WVU to assist with the design and analysis of control and navigation laws for UAV during nominal and abnormal autonomous flight conditions [82]. This computational package was built in MATLAB®/Simulink® for maximum flexibility and portability and includes several major modules such as aircraft models, trajectory generation algorithms, trajectory tracking control laws, actuator failure models, sensor failure model, GPS model, and atmospheric adverse phenomena. An open source freely available simulation package, FlightGear, displays visual cues of the aircraft and environmental scenery. An additional customized visualization tool allows for interactive mission scenario setup based on desired waypoints, 3-dimensional obstacles, and risk zones. User-friendly graphical interface menus can be used for the setup of a primary simulation scenario elements. The main portal of the WVU UAS interface is presented in Figure 11. The FlightGear and customized interface for monitoring aircraft and flight path are presented in Figure 12.

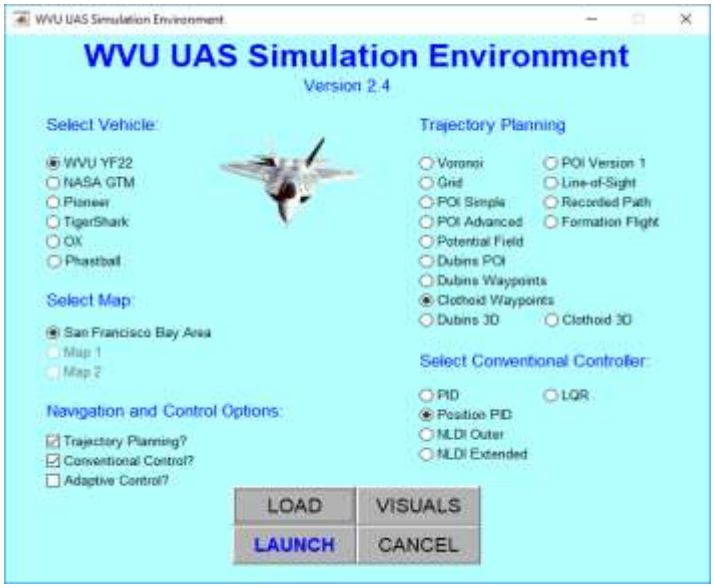

Figure 11. WVU UAS simulation figure environment mission scenario setup menu.

Figure 12. WVU UAS simulation environment–visualization interfaces.

For this research, the WVU YF22 UAV model was used. The commanded trajectory was generated using Dubins or clothoid algorithms, while the trajectory tracking depended on the control laws [82]. The Inertial Measurement Unit (IMU) is a prebuilt block of Simulink's aerospace block-set. The IMU model consists of three-axis accelerometer and three-axis gyroscope. The errors model for the IMU components is shown in Figure 13.

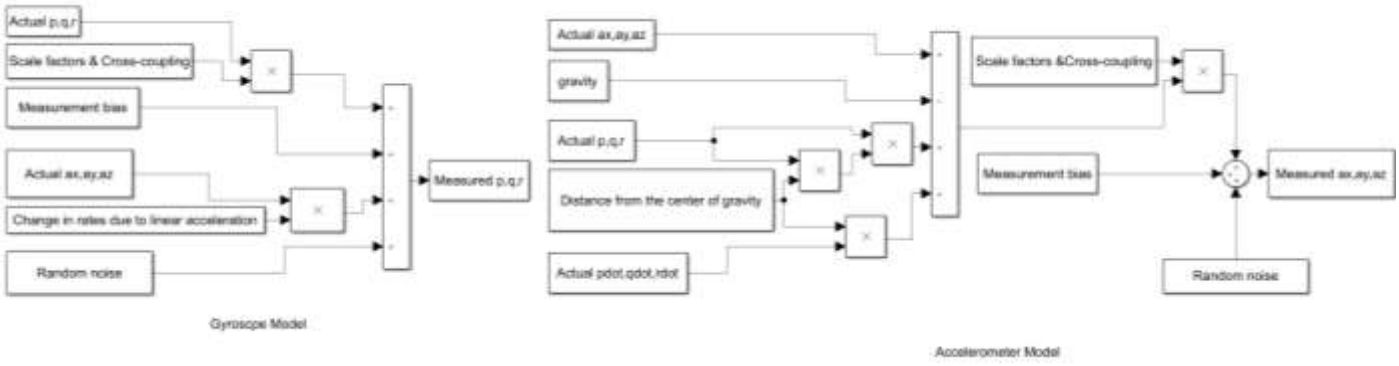

Figure 13. Errors model for the IMU components (accelerometer and gyroscope).

5. Example of Framework Implementation

5.1. Baseline Implementation

For the successful implementation of the proposed AIS framework, the most critical elements include the selection of UAV features φ_s, correction features φ_c, antigen matching algorithm, and estimation correction algorithm $\mathcal{F}$.

5.1.1. The UAV Features

For this research, UAV features were the normalized gradients in body axes of the measured translational acceleration and angular velocity:

$$\varphi_s = \{\Delta a_{xm}\, \Delta a_{ym}\, \Delta a_{zm}\, \Delta p_m\, \Delta q_m\, \Delta r_m\} \tag{30}$$

and:

$$\Delta a_{xm}^{(k)} = a_{xm}^{(k)} - a_{xm}^{(k-1)} \tag{31}$$

$$\Delta a_{ym}^{(k)} = a_{ym}^{(k)} - a_{ym}^{(k-1)} \tag{32}$$

$$\Delta a_{zm}^{(k)} = a_{zm}^{(k)} - a_{zm}^{(k-1)} \tag{33}$$

$$\Delta p_m^{(k)} = p_m^{(k)} - p_m^{(k-1)} \tag{34}$$

$$\Delta q_m^{(k)} = q_m^{(k)} - q_m^{(k-1)} \tag{35}$$

$$\Delta r_m^{(k)} = r_m^{(k)} - r_m^{(k-1)} \tag{36}$$

Such measurements are used to generate the AIS as a set of AMCs; therefore, with the sampling approach, an AMC M_j is denoted as:

$$M_j = \left[\Delta a_{xj} \;\; \Delta a_{yj} \;\; \Delta a_{zj} \;\; \Delta p_j \;\; \Delta q_j \;\; \Delta r_j \;\; \vdots \;\; \varphi_{cj}\right] \tag{37}$$

For a more compact notation, the following definitions will also be used:

$$[\Delta\vec{\omega}_m]_B = \begin{bmatrix}\Delta p_m \\ \Delta q_m \\ \Delta r_m\end{bmatrix}_B \tag{38}$$

$$[\Delta\vec{a}_m\,]_E = \begin{bmatrix}\Delta a_{xm} \\ \Delta a_{ym} \\ \Delta a_{zm}\end{bmatrix}_E \tag{39}$$

$$[\Delta\vec{\omega}]_B = \begin{bmatrix}\Delta p \\ \Delta q \\ \Delta r\end{bmatrix}_B \tag{40}$$

$$[\Delta\vec{a}\,]_E = \begin{bmatrix}\Delta a_x \\ \Delta a_y \\ \Delta a_z\end{bmatrix}_E \tag{41}$$

and:

$$M_j = \left[\left[\Delta\vec{a}_j\,\right]_E \left[\Delta\vec{\omega}_j\right]_B \vdots\, \varphi_{cj}\right] \tag{42}$$

5.1.2. The Acceleration and Angular Rate Measurements

The acceleration and angular rate measurements were simulated using an inertial measurement unit (IMU) model shown in Figure 13. The acceleration body axes components from the IMU are rotated and integrated twice to estimate the velocity and the position components in the inertial CS.

5.1.3. The Position and Velocity Estimation

The position and velocity estimation algorithm consisted of merely integrating the outputs from the IMU. Equation 43 is the estimation of velocity of the vehicle and equation 44 is the estimation of the position of the vehicle.

$${}^E\vec{v}_e^{\,O} = \int {}^E\vec{a}_m^{\,O} dt \tag{43}$$

$$\vec{p}_e^{\,EO} = \iint {}^E\vec{a}_m^{\,O}\, dt \tag{44}$$

5.1.4. The Correction Features

Two cases were considered to build the AIS; correction of the outputs of the estimation scheme represented by the velocity and position [83], and correction of the input of the estimation scheme represented by the acceleration [84].

5.1.5. The Matching Algorithm

The matching algorithm involves identifying a good match between the current UAV features and the AIS antigens using immune affinity distance which is the measured distance of the degree of the interaction. Each current sample of UAV features or antigen is checked against the UAV features in the AIS. The index of the antigen that is matched with the UAV features will be used to extract the corresponding correction or antibody from the AIS. The antigen for correction of estimation scheme input and output is the same and represented by the acceleration components and the angular rates. Maximum affinity, which corresponds to the minimum Manhattan distance, is used to determine the index of the memory cell from which the antibody is extracted; however, the normalization can only be applied to the antigen part of the AMC.

The normalized memory cell $\bar{M}_j$ is:

$$\bar{M}_j = \left[\Delta\bar{\mathrm{a}}_{\mathrm{xj}}\ \Delta\bar{\mathrm{a}}_{\mathrm{yj}}\ \Delta\bar{\mathrm{a}}_{\mathrm{zj}}\ \Delta\bar{\mathrm{p}}_{\mathrm{j}}\ \Delta\bar{\mathrm{q}}_{\mathrm{j}}\ \Delta\bar{\mathrm{r}}_{\mathrm{j}} \vdots \varphi_{cj}\right] \tag{45}$$

Alternatively,

$$\bar{M}_j = \left[\bar{\varphi}_{sjn} \vdots \varphi_{cj}\right], \quad n = 1,2,\dots,6 \tag{46}$$

The normalized UAV features are defined as:

$$\bar{\varphi}_{sjn} = \frac{\varphi_{sjn} - \min_j(\varphi_{sjn})}{\max_j(\varphi_{sjn}) - \min_j(\varphi_{sjn})} \tag{47}$$

The current antigen is normalized over the same range:

$$\bar{\varphi}_{sn}^{(\propto)} = \frac{\varphi_{sn}^{(\propto)} - \min_j(\varphi_{sjn})}{\max_j(\varphi_{sjn}) - \min_j(\varphi_{sjn})} \tag{48}$$

The affinity metric is first calculated for each system feature of the current antigen as the Manhattan distance (MD_{jn}) to all antigens in the AIS:

$$MD_{jn} = \left| \bar{\varphi}_{sjn} - \bar{\varphi}_{sn}^{(\propto)} \right| \tag{49}$$

Then:

$$MD_j = \sum_{n=1}^{N_M} MD_{jn} \tag{50}$$

The identified artificial memory cell to be used for compensation extraction is the one for which MD_j is minimal. The UAV features can be expanded while different features could be considered to improve the accuracy of the results; for example, the angular rate components. The process of immune affinity for antibody extraction is illustrated in Figure 14 and is applied for both corrections of estimation scheme input and output. This mechanism consists of the Manhattan distance between the normalized gradient of the UAV features at each time step and the normalized antigen.

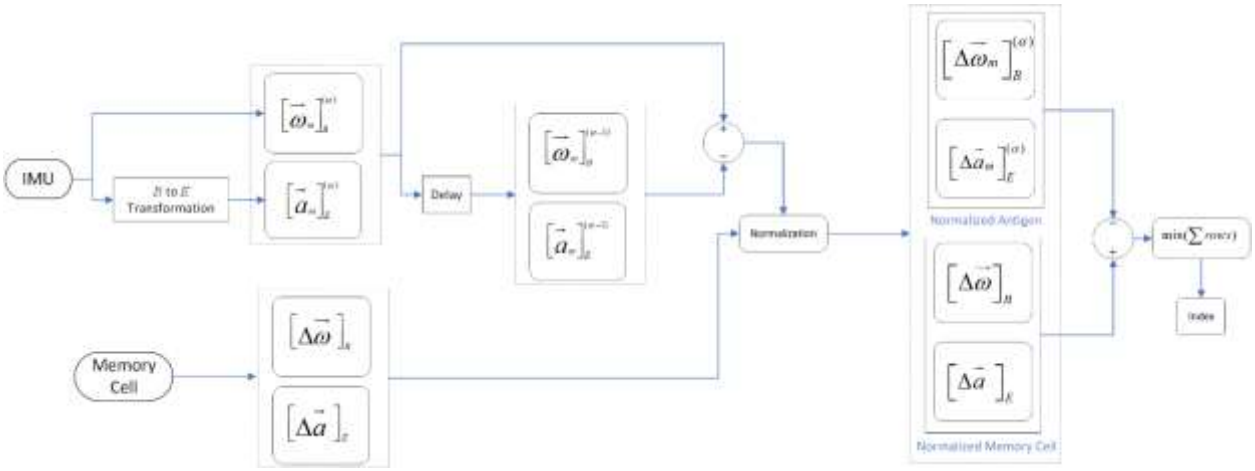

Figure 14. Immune affinity for antibody extraction.

5.2. Correction of Estimation Scheme Output

5.2.1. AIS Generation

The correction features were obtained from an integral estimation scheme and simulated GPS measurements at nominal conditions. The integration errors at each instant are expected to be

corrected using proper antibodies from the memory cells, as identified through the matching algorithm. The correction features are represented as the normalized differences between the gradients of the estimated and measured vehicle position and velocity in the inertial CS:

$$\varphi_c = \{\Delta x \ \Delta y \ \Delta z \ \Delta v_x \ \Delta v_y \ \Delta v_z\} \tag{51}$$

With the sampling approach, a memory cell consists of values at generation samples *k*, expressed as:

$$M_j = [\Delta a_{xj} \ \Delta a_{yj} \ \Delta a_{zj} \ \Delta p_j \ \Delta q_j \ \Delta r_j \vdots \Delta x_j \ \Delta y_j \ \Delta z_j \ \Delta v_{xj} \ \Delta v_{yj} \ \Delta v_{zj}] \tag{52}$$

where the expression for the UAV features are given in (30) through (32), and:

$$[\vec{p}_e]_E = \begin{bmatrix} x_e \\ y_e \\ z_e \end{bmatrix}_E \tag{53}$$

$$[\vec{v}_e]_E = \begin{bmatrix} v_{xe} \\ v_{ye} \\ v_{ze} \end{bmatrix}_E \tag{54}$$

$$[\vec{p}_m]_E = \begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix}_E \tag{55}$$

$$[\vec{v}_m]_E = \begin{bmatrix} v_{xm} \\ v_{ym} \\ v_{zm} \end{bmatrix}_E \tag{56}$$

$$[\Delta\vec{p}]_E = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}_E \tag{57}$$

$$[\Delta\vec{v}]_E = \begin{bmatrix} \Delta v_x \\ \Delta v_y \\ \Delta v_z \end{bmatrix}_E \tag{58}$$

$$\boldsymbol{\Delta x^{(k)}} = \left[\left(x_e^{(k)} - x_e^{(k-1)}\right) - \left(x_m^{(k)} - x_m^{(k-1)}\right)\right] \tag{59}$$

$$\boldsymbol{\Delta y^{(k)}} = \left[\left(y_e^{(k)} - y_e^{(k-1)}\right) - \left(y_m^{(k)} - y_m^{(k-1)}\right)\right] \tag{60}$$

$$\boldsymbol{\Delta z^{(k)}} = \left[\left(z_e^{(k)} - z_e^{(k-1)}\right) - \left(z_m^{(k)} - z_m^{(k-1)}\right)\right] \tag{61}$$

$$\boldsymbol{\Delta v_x^{(k)}} = \left[(v_{xe}^{(k)} - v_{xe}^{(k-1)}) - (v_{xm}^{(k)} - v_{xm}^{(k-1)})\right] \tag{62}$$

$$\boldsymbol{\Delta v_y^{(k)}} = \left[(v_{ye}^{(k)} - v_{ye}^{(k-1)}) - (v_{ym}^{(k)} - v_{ym}^{(k-1)})\right] \tag{63}$$

$$\boldsymbol{\Delta v_z^{(k)}} = \left[(v_{ze}^{(k)} - v_{ze}^{(k-1)}) - (v_{zm}^{(k)} - v_{zm}^{(k-1)})\right] \tag{64}$$

The block diagram for building the AIS that consists of memory cells is presented in Figure 15, where $\vec{\omega}$ is the angular rates vector, $\vec{a}$ is the acceleration vector, $\vec{p}$ the position vector and $\vec{v}$ is the velocity vector. The subscript *m* defines the measured vectors, while the subscript *e* defines the estimated vectors while the superscript k and k-1 refer to the sampling time:

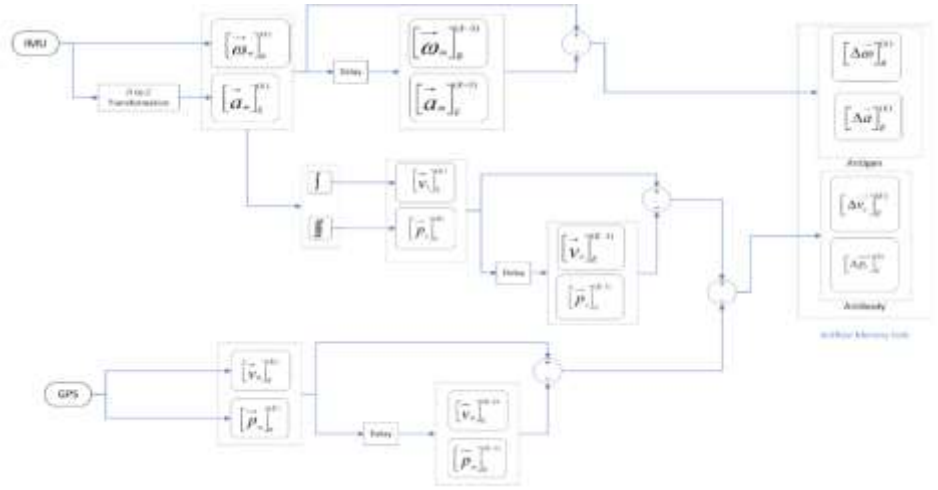

Figure 15. Generating AIS as a collection of artificial memory cells including antibodies of velocity and position.

5.2.2. AIS-based Estimation Correction

The antibody representing the errors of the change in position and velocity components at the current time step (φ_{cj}^{α}) can now be extracted and used to compensate the output of the estimation scheme as illustrated in Figure 16. The closed-loop UAV system with AIS correction

is illustrated in Figure 17. Assuming the position and velocity estimates at time step ∝ as $E^{(\propto)}$ and the corrected estimates as $E_c^{(\propto)}$ then:

$$E_c^{(\propto)} = E^{(\propto)} + \varphi_{cj}{}^{(\alpha)} \tag{65}$$

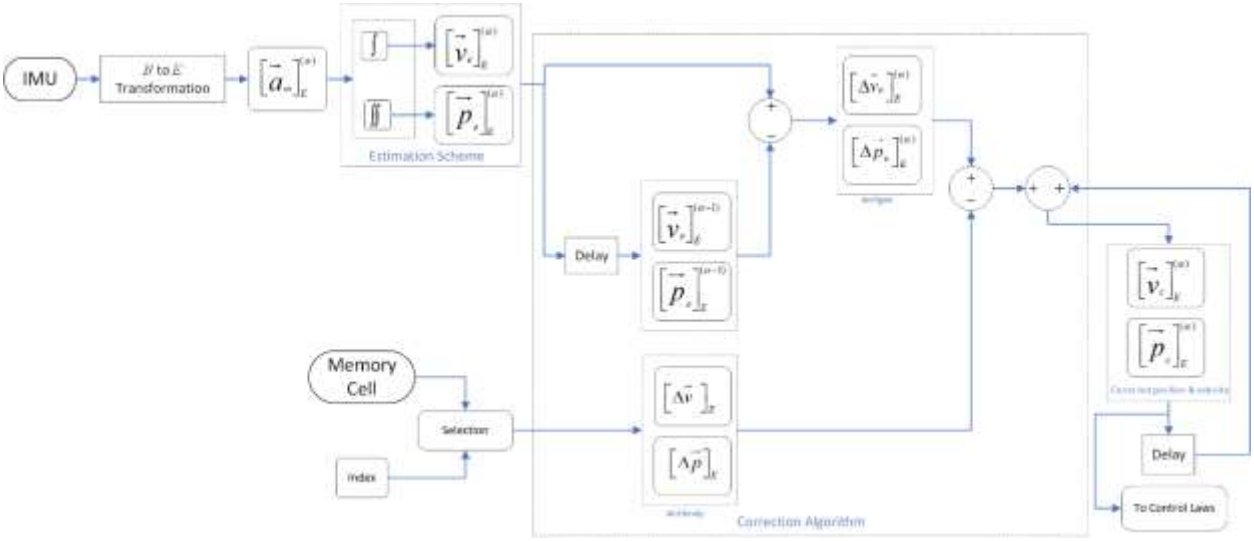

Figure 16.Correction of the estimated position and the velocity components.

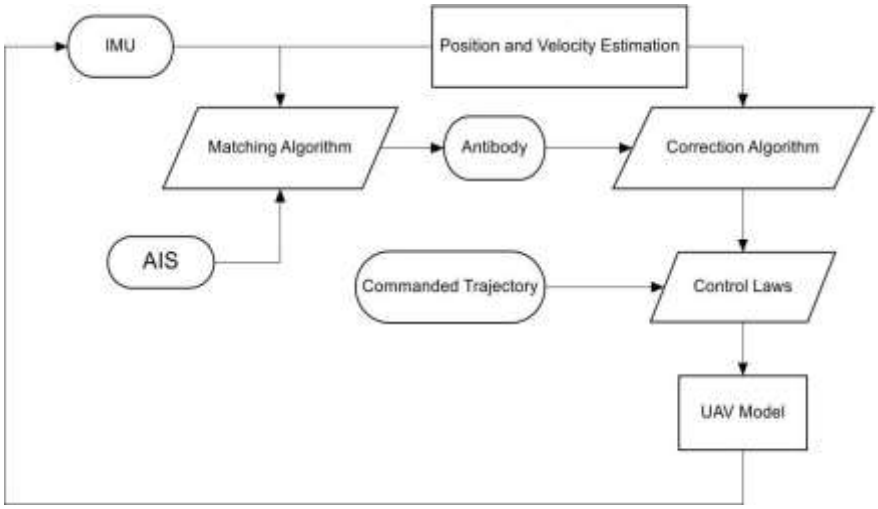

Figure 17. Closed loop UAV system with AIS correction of position and velocity.

5.3. Correction of Estimation Scheme Input

5.3.1. AIS Generation

The UAV features were obtained from the IMU and simulated GPS measurements at nominal conditions. The position and velocity estimation algorithm consisted of integrating the

corrected acceleration. The acceleration and angular rate measurements were simulated using the IMU model:

$$\varphi_s = \{\Delta a_{xm}\ \ \Delta a_{ym}\ \ \Delta a_{zm}\ \ \Delta p_m\ \ \Delta q_m\ \ \Delta r_m\} \tag{66}$$

The correction features were the differences between the vehicle acceleration estimated from the GPS measurements and the vehicle acceleration measured from the IMU in the inertial coordinate system CS:

$$\varphi_c = \{\Delta a_{xc}\ \ \Delta a_{yc}\ \ \Delta a_{zc}\} \tag{67}$$

Errors in measured acceleration at each instant are expected to be corrected using proper antibodies from the memory cells. The acceleration body axes components from the correction scheme are integrated twice and rotated to estimate the velocity and the position components in the inertial CS. The block diagram of building the AIS consisting of memory cells is presented in Figure 18.

Based on the sampling approach, a memory cell consists of values at generation samples *k*, expressed as:

$$M_j = [\Delta a_{xj}\ \ \Delta a_{yj}\ \ \Delta a_{zj}\ \ \Delta p_j\ \ \Delta q_j\ \ \Delta r_j\ \ \vdots\ \ a_{xcj}\ \ a_{ycj}\ \ a_{zcj}\ ] \tag{68}$$

where the UAV features are the same as for the previous approach. For compactness, the following notation will be used in the block diagrams:

$$[\vec{v}_m\ ]_E = \begin{bmatrix} v_{xm} \\ v_{ym} \\ v_{zm} \end{bmatrix}_E \tag{69}$$

$$[\vec{a}_e\ ]_E = \begin{bmatrix} a_{xe} \\ a_{ye} \\ a_{ze} \end{bmatrix}_E \tag{70}$$

$$[\vec{a}_m\,]_E = \begin{bmatrix} a_{xm} \\ a_{ym} \\ a_{zm} \end{bmatrix}_E \tag{71}$$

$$[\vec{\omega}_m]_B = \begin{bmatrix} \Delta p_m \\ \Delta q_m \\ \Delta r_m \end{bmatrix}_B \tag{72}$$

$$[\vec{a}_c]_E = \begin{bmatrix} a_{xc} \\ a_{yc} \\ a_{zc} \end{bmatrix}_E \tag{73}$$

$$a_{xc} = a_{xm} - a_{xe} \tag{74}$$

$$a_{yc} = a_{ym} - a_{ye} \tag{75}$$

$$a_{zc} = a_{zm} - a_{ze} \tag{76}$$

where:

$$[\vec{a}_e]_E = \frac{d[\vec{v}_m]_E}{dt} \tag{77}$$

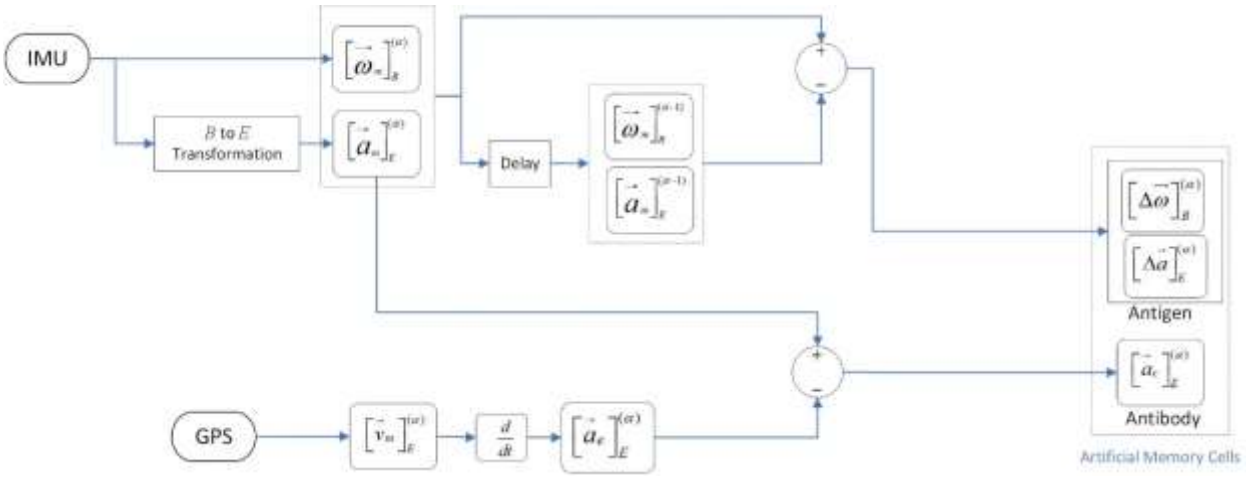

Figure 18. Generating AIS as a collection of artificial memory cells including antibodies of acceleration.

5.3.2. AIS-Based Estimation Correction

The antibody representing the errors of the change in acceleration components at the current time step ($\varphi_{cj}^{(\alpha)}$) can now be extracted and used to compensate for the estimation scheme, as illustrated in Figure 19. Let us denote the estimated acceleration at time step $\propto$ as $E^{(\propto)}$ and the corrected acceleration as $E_c^{(\propto)}$ then:

$$E_c^{(\propto)} = E^{(\propto)} + \varphi_{cj}^{(\alpha)} \tag{78}$$

The corrected estimated values of vehicle position and velocity in inertial coordinates are used instead of absent GPS output, as illustrated in Figure 20.

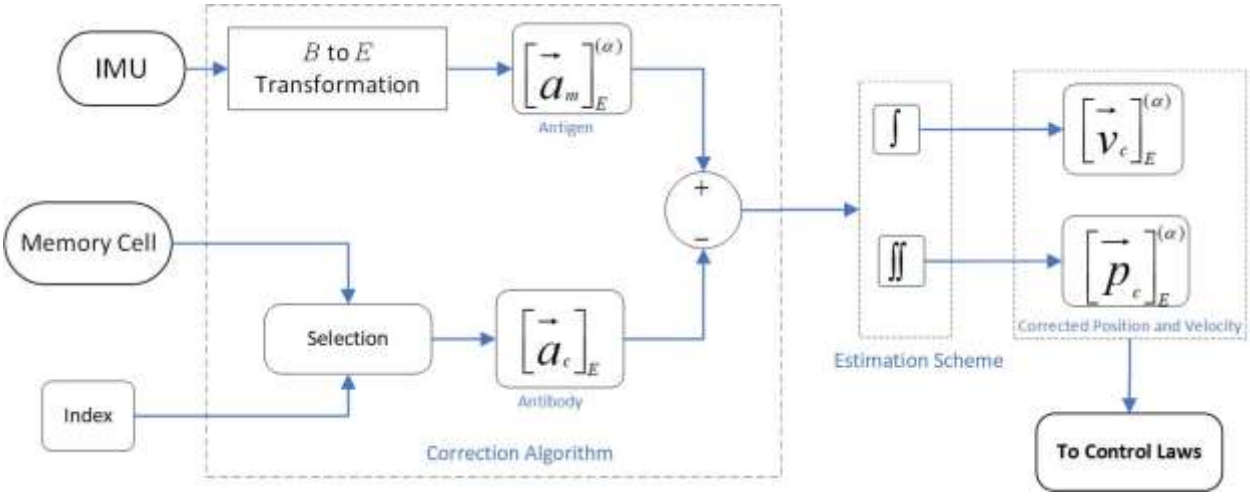

Figure 19. Correction of the acceleration components.

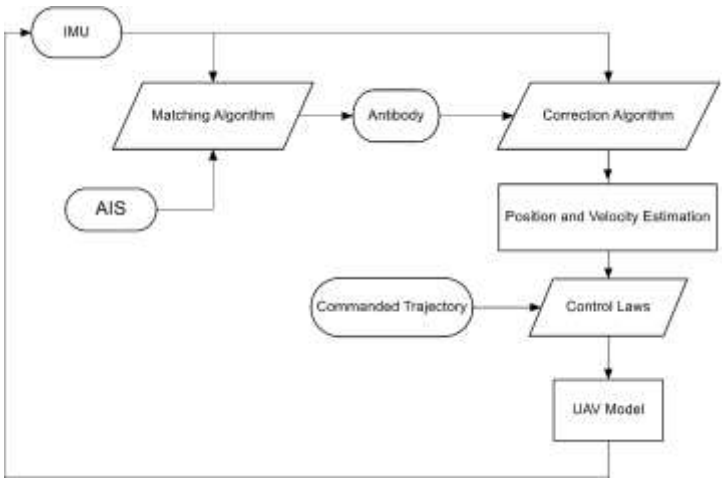

Figure 20. Closed loop UAV system with AIS correction of acceleration.

6. Testing and Performance Evaluation

6.1. Experimental Design

Two approaches were used to implement the AIS. One approach, referred to as the "output approach," relies on correcting the output of the estimation scheme, that is the estimation of vehicle position and velocity, as addressed in section 6.2. The other approach, referred to as "input approach" relies on correcting the input of the estimation scheme, that is the measured acceleration as addressed in section 6.3. Six arbitrary trajectories, as presented in Figures 21-26, were simulated using the WVU UAS simulation environment, and extensive simulation data were recorded for creating the AIS artificial memory cells. These trajectories are referred to as AIS generation trajectories. Note that the nominal conditions under which the AIS generation trajectories are obtained, constitute flight scenarios that are representative for the dynamics of the system, with all systems functioning properly as designed, as per the definition of the self within the immunity paradigm.

The vehicle features were the same for both approaches, while the correction features differed. Two additional commanded trajectories were built for AIS validation purposes while considering three scenarios for each commanded trajectory. The nominal or normal conditions scenario comprises the availability of vehicle position and velocity measurements from well operating GPS or other equivalent sources for tracking the commanded trajectory. It is also assumed that outputs are available from an onboard vehicle position and estimation scheme using IMU measurements such that the values of AIS correction features can be determined. The estimation scheme only scenario assumes that measurements of vehicle position and velocity are not available from GPS or other sources and the autonomous trajectory tracking relies only on outputs from the estimation scheme with the IMU operating at the same level of performance as during the nominal condition scenario.

The AIS scenario assumes that vehicle position and velocity measurements are not available and that the commanded trajectory is tracked using estimation scheme outputs corrected with AIS antibodies. The metrics for performance evaluation [7] of the proposed methodology rely on trajectory tracking errors and are defined as a maximum error, mean of the errors, and standard deviation. These measures must be evaluated for the horizontal plane error (X, Y), vertical error (Z), and total error (X, Y, Z). The trajectory tracking errors (e) are

calculated as the difference between the commanded position value from path planning and the actual position value from simulation results. The vector of evaluation metrics for the trajectory tracking performance can be expressed as:

$$\boldsymbol{PV_{TT}} = [\bar{\boldsymbol{e}}_{\boldsymbol{maxXY}}\ \bar{\boldsymbol{e}}_{\boldsymbol{maxZ}}\ \bar{\boldsymbol{e}}_{\boldsymbol{maxXYZ}}\ \bar{\boldsymbol{e}}_{\boldsymbol{XY}}\ \bar{\boldsymbol{e}}_{\boldsymbol{Z}}\ \bar{\boldsymbol{e}}_{\boldsymbol{XYZ}}\ \hat{\boldsymbol{e}}_{\boldsymbol{XY}}\ \hat{\boldsymbol{e}}_{\boldsymbol{Z}}\ \hat{\boldsymbol{e}}_{\boldsymbol{XYZ}}]^{\boldsymbol{T}} \qquad (78)$$

where:

$\bar{e}_{maxXY}$: Maximum of the XY plane trajectory tracking error.

$\bar{e}_{maxZ}$: Maximum of the vertical trajectory tracking error.

$\bar{e}_{maxXYZ}$: Maximum of the combined XYZ trajectory tracking error.

$\bar{e}_{XY}$: Mean of the combined XY trajectory tracking error.

$\bar{e}_{Z}$: Mean of the vertical trajectory tracking error.

$\bar{e}_{XYZ}$: Mean of the combined XYZ trajectory tracking error.

$\hat{e}_{XY}$: Standard deviation of the XY plane trajectory tracking error.

$\hat{e}_{Z}$: Standard deviation of the vertical trajectory tracking error.

$\hat{e}_{XYZ}$: Standard deviation of the combined XYZ trajectory tracking error.

The effect of navigation accuracy on AIS is discussed in section 6.4. The difference between two trajectory path planning, clothoid and Dubins, is implemented in section 6.5. The IMU sensitivity and grade is investigated in section 6.6 and identify the IMU used with the AIS. In section 6.7, two affinity approaches are compared; Manhattan Affinity and Euclidean Affinity. The features performance or the effect of each feature in the AIS selection to the best index is investigated in section 6.8, and finally, the augmentation of position and vertical velocity components is presented in section 6.9.

6.2. Correction of Estimation Scheme Output

The six AIS generation trajectories are presented in Figures 21- 26. The values of the UAV features φ_s and correction features φ_c were collected for building an AIS capable of storing corrections to the output of the estimation scheme (vehicle position and velocity). Each

AIS generation commanded trajectory had a different duration and included constant altitude and speed commands. All systems, including GPS, IMU, and estimation scheme were operating under normal conditions. All data from the six trajectories were collected and merged to produce one set of artificial memory cells.

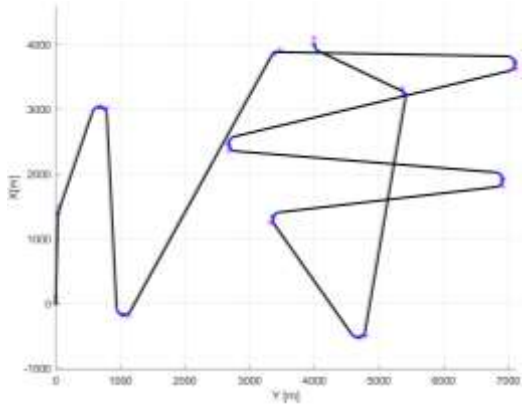

Figure 21. AIS generation trajectory #1.

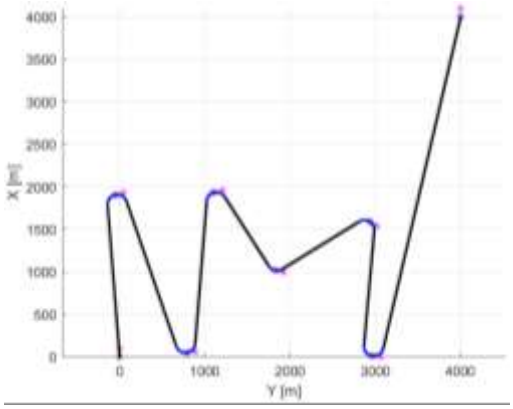

Figure 22.AIS generation trajectory #2.

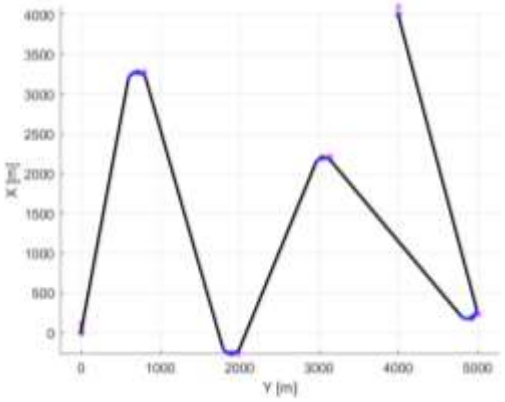

Figure 23. AIS generation trajectory #3.

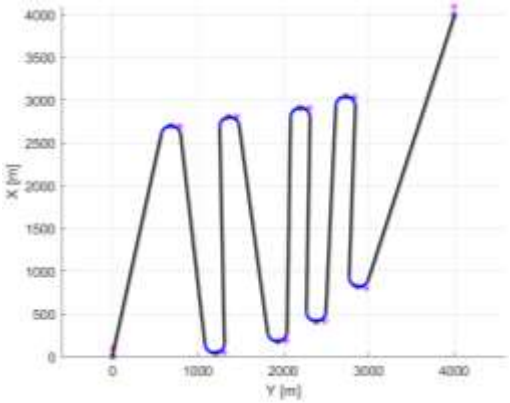

Figure 24. AIS generation trajectory #4.

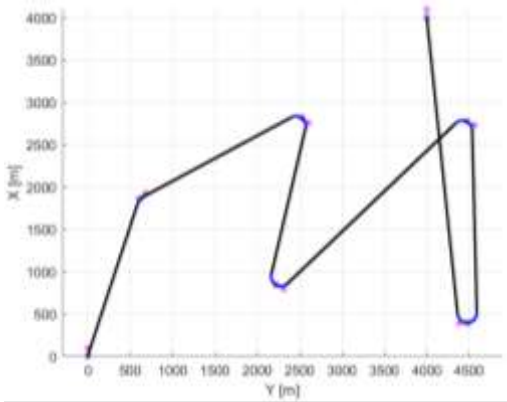

Figure 25. AIS generation trajectory #5.

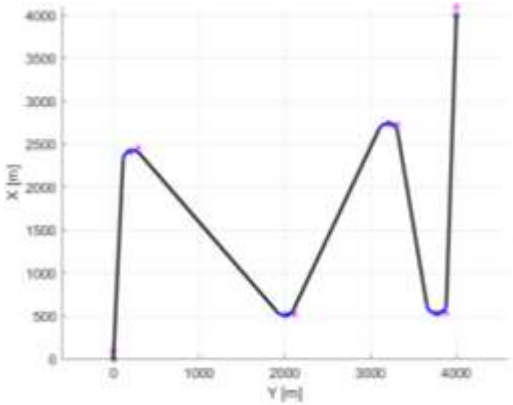

Figure 26. AIS generation trajectory #6.

Once the AIS was built, the AIS generation trajectories were used for verification. All tests showed similar trends. The trajectory tracking performance under normal conditions was

expectedly good. The trajectory tracking under the estimation-scheme-only scenario exhibited significant error accumulation. Finally, the third testing scenario showed that the AIS could be used successfully to provide corrections to the estimation scheme and achieve trajectory tracking performance similar to the normal condition case. For illustration, the results obtained for the AIS generation trajectory #2 are presented in Figure 27 for normal conditions, in Figure 29 for the estimation-scheme-only scenario, and in Figure 31 for the AIS scenario. The normalized tracking errors under the three scenarios are presented in Figure 36.

The normalization factor was the maximum trajectory tracking error recorded under nominal conditions. The percentage improvements of trajectory tracking errors for relevant metrics, relative to the nominal conditions' scenario are summarized in Table 1 and range between 86% and 92%. Two different trajectories were used for validation. The tracking of validation trajectory #1 and #2 under the nominal conditions' scenario are presented in Figures 28 and 33, respectively. As expected, the tracking performance was extremely good. The data from the validation trajectories under nominal condition have not been used for AIS generation. Next, the two validation trajectories were tested when the GPS was absent, and the estimation scheme output substituted its output. The tracking of the two validation trajectories under the estimation-scheme-only scenario is presented in Figures 30 and 35, respectively.

The input of the estimation scheme was the IMU output which is the measured accelerations. The plots illustrate the effect of the cumulative errors from the integrated measured accelerations on the trajectory tracking. It is evident that, in both cases, significant tracking error accumulations occur. Under the AIS scenario, the GPS output was substituted by the estimation scheme output corrected by the AIS antibodies which were extracted from the artificial memory cells with the assistance of the matching algorithm. Figures 32 and 33 present the tracking of the commanded validation trajectories #1 and #2 under the AIS scenario, respectively. In both cases, it can be seen that the tracking error accumulation has been significantly reduced as compared to the estimation-scheme-only scenario.

The normalized tracking errors under the three scenarios for the validation trajectory #1 are presented in Figure 37 and similar results for validation trajectory #2 are presented in Figure 38. The percentage improvements of trajectory tracking errors for relevant metrics relative to the

nominal conditions scenario for both validation trajectories are summarized in Table 1. Overall, these improvements range between 25% and 81%. Significant improvements in the trajectory tracking errors can be observed when using the AIS approach as compared to using the estimation scheme alone. Promising performance is recorded for both the generation and validation data. The percentages change in errors for the nominal errors are illustrated in Tables 2-4. All these tables show that the AIS scenario can reduce the errors produced by the estimation-scheme-only scenario significantly.

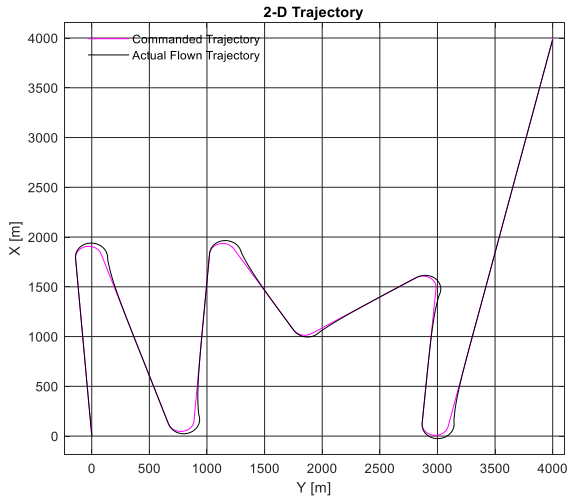

Figure 27. Tracking of AIS generation trajectory under nominal conditions scenario.

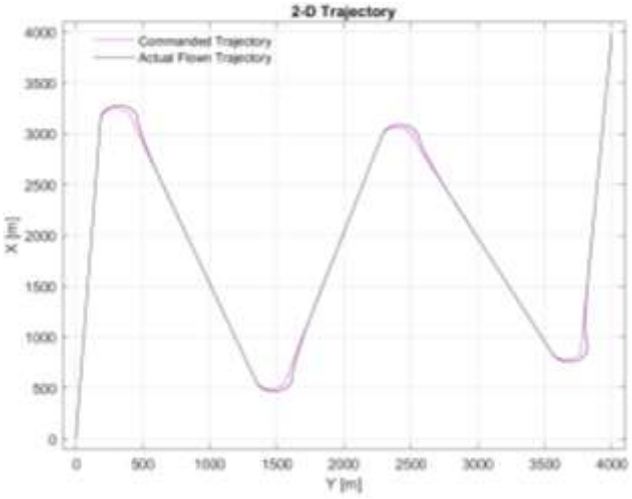

Figure 28. Tracking of validation trajectory #1 under nominal conditions scenario.

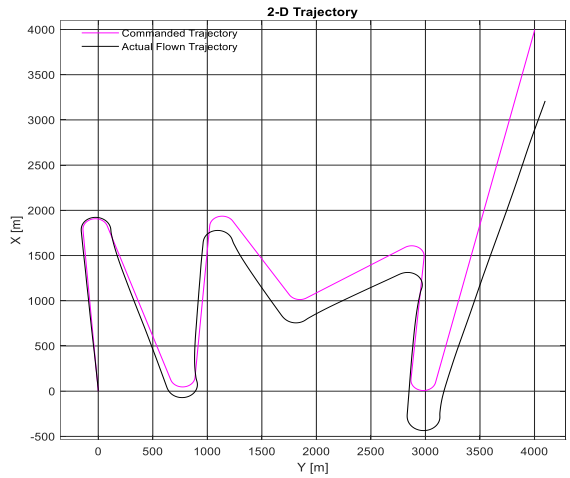

Figure 29. Tracking of AIS generation trajectory with estimation scheme only.

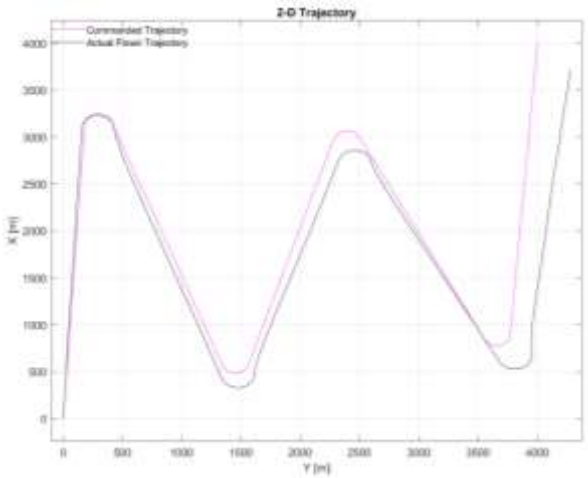

Figure 30. Tracking of validation trajectory #1 with estimation scheme only.

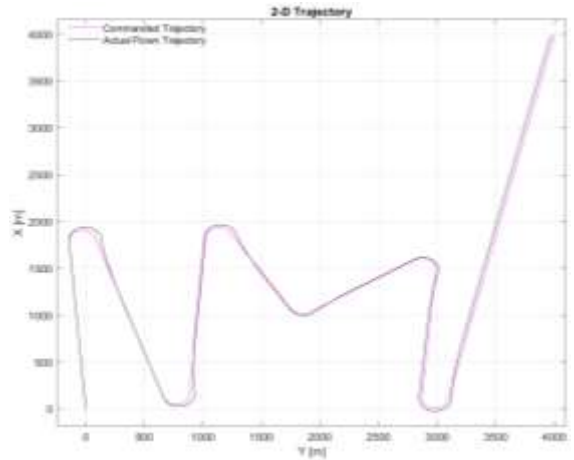

Figure 31. Tracking of AIS generation trajectory with AIS corrected estimation.

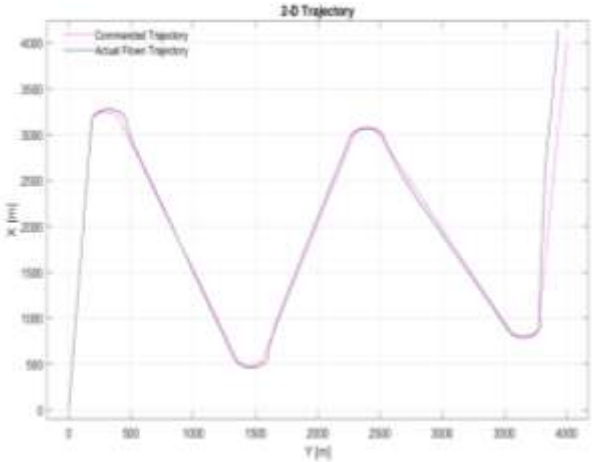

Figure 32. Tracking of validation trajectory #1 with AIS corrected estimation.

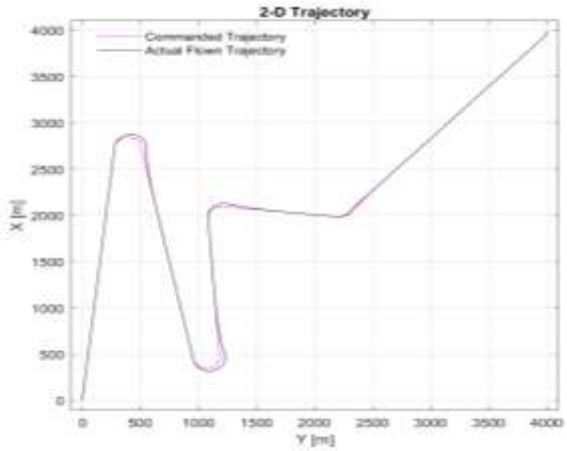

Figure 33. Tracking of validation trajectory #2 under nominal conditions scenario.

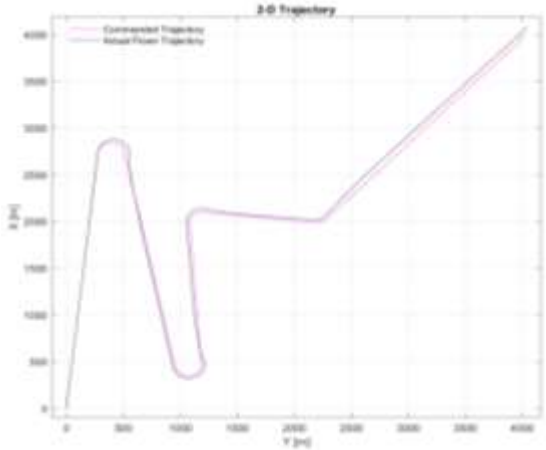

Figure 34. Tracking of validation trajectory #2 with AIS corrected estimation.

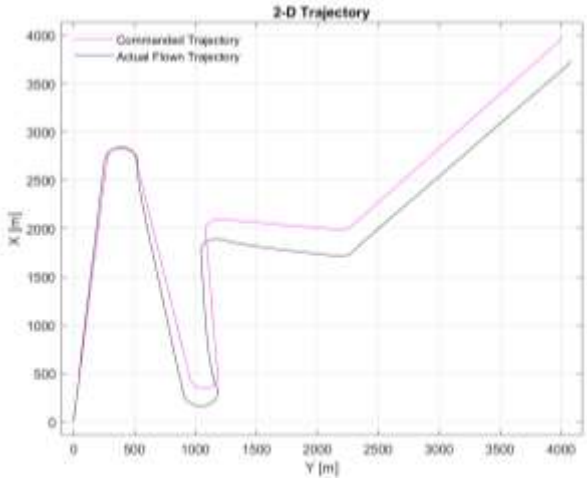

Figure 35. Tracking of validation trajectory #2 with estimation scheme only.

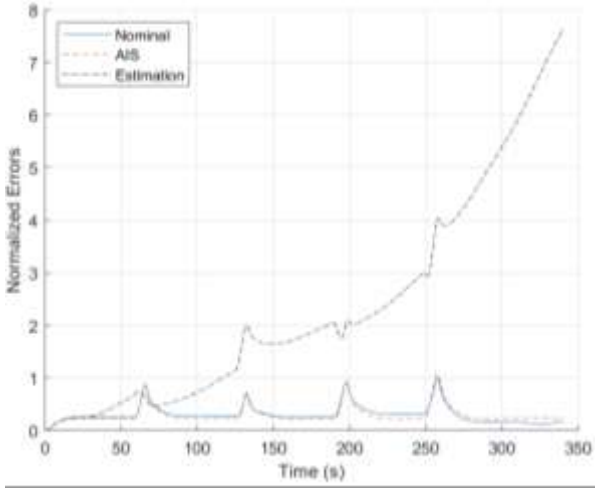

Figure 36. Normalized tracking errors of AIS generation trajectory.

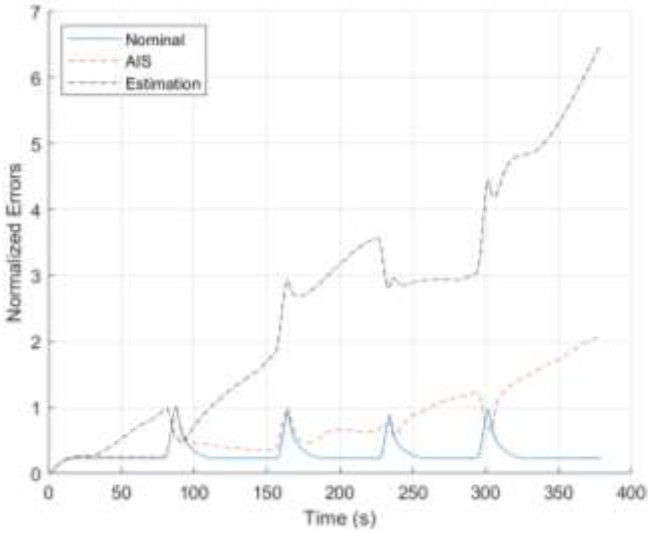

Figure 37. Normalized tracking errors of validation trajectory #1.

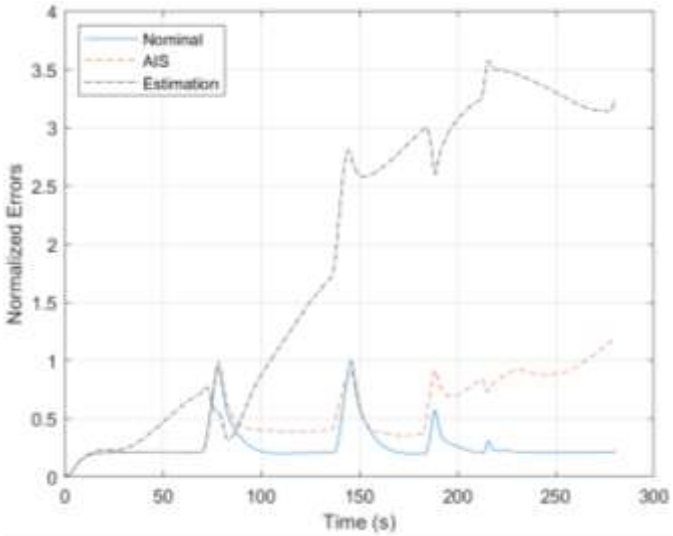

Figure 38. Normalized errors of validation trajectory #2.

Table 1. The percentage improvement of trajectory tracking under the AIS scenario.

Errors [%]	AIS Generation Trajectory	Validation Trajectory #1	Validation Trajectory #2
Max XY Error [%]:	87	60	69
Max Z Error [%]:	88	70	30
Max XYZ Error [%]:	87	67	67
Mean XY Error [%]:	86	72	74
Mean Z Error [%]:	92	67	41
Mean XYZ Error [%]:	87	70	72
Standard Deviation XY Error [%]:	92	72	81
Standard Deviation Z Error [%]:	90	63	25
Standard Deviation XYZ Error [%]:	92	70	77

Table 2. Percentage change in errors for AIS generation trajectory with respect to nominal errors.

Errors	AIS	Estimation
Max XY Error [%]:	6	608
Max Z Error [%]:	14	571
Max XYZ Error [%]:	8	603
Mean XY Error [%]:	5	555
Mean Z Error [%]:	30	1513
Mean XYZ Error [%]:	3	613
Standard Deviation XY Error [%]:	4	984
Standard Deviation Z Error [%]:	12	986
Standard Deviation XYZ Error [%]:	17	947

Table 3. Percentage change in errors for validation trajectory #1 with respect to nominal errors.

Errors	AIS	Estimation
Max XY Error [%]:	107	423
Max Z Error [%]:	367	1471
Max XYZ Error [%]:	109	549
Mean XY Error [%]:	120	676
Mean Z Error [%]:	944	3048
Mean XYZ Error [%]:	160	762
Standard Deviation XY Error [%]:	179	873
Standard Deviation Z Error [%]:	612	1790
Standard Deviation XYZ Error [%]:	120	975

Table 4. Percentage change in errors for validation trajectory #2 with respect to nominal errors.

Generation	AIS	Estimation
Max XY Error [%]:	13	256
Max Z Error [%]:	172	288
Max XYZ Error [%]:	19	257
Mean XY Error [%]:	80	589
Mean Z Error [%]:	597	1079
Mean XYZ Error [%]:	100	598
Standard Deviation XY Error [%]:	59	712
Standard Deviation Z Error [%]:	54	501
Standard Deviation XYZ Error [%]:	89	699

6.3. Correction of Estimation Scheme Input

An alternative solution to correcting the position and velocity estimates of the acceleration integration scheme involves adjusting the acceleration measurement (input to the estimation scheme), such that the resulting position and velocity estimates are accurate. The implementation of this approach considers the same set of vehicle characteristics, but different correction features as described in previous sections. The AIS was built using the same six generation trajectories as for the output approach. All verification tests based on generation trajectories showed similar trends. The trajectory tracking performance under normal conditions was good as expected. The trajectory tracking under the estimation-scheme-only scenario exhibited significant error accumulation. Finally, the third testing scenario showed that the AIS could be used successfully to provide corrections to the estimation scheme and achieve trajectory tracking performance similar to the normal condition case.

For illustration, the results obtained for the AIS generation trajectory #1 are presented in Figure 39 for normal conditions, in Figure 40 for the estimation-scheme-only scenario, and in Figure 41 for the AIS scenario. The normalized tracking errors under the three scenarios are presented in Figure 44.

The percentage improvements of trajectory tracking errors for relevant metrics relative to the nominal conditions' scenario are summarized in Table 5 and vary between 90% and 95%. The same two different trajectories as with the output approach were used for validation. The tracking of validation trajectory #1 and #2 under the nominal conditions' scenario are presented in Figures 28 and 33, respectively. Next, the two validation trajectories were tested when the GPS was absent, and the estimation scheme output substituted its output. The tracking of the two validation trajectories under the estimation-scheme-only scenario are presented in Figures 30 and 35, respectively. The input of the estimation scheme was the IMU output; the measured accelerations. The plots illustrate the effect of the cumulative errors from the integrated measured accelerations on the trajectory tracking. It can be seen that significant tracking error accumulations occur in both cases. Under the AIS scenario, the acceleration components as an output of the IMU were corrected by the AIS antibodies, which were extracted from the artificial memory cells with the assistance of the matching algorithm, and then used by the estimation scheme input to estimate the position and velocity components. Figures 42 and 43 present the

tracking of the commanded validation trajectories #1 and #2 under the AIS scenario, respectively. In both cases, it can be seen that the tracking error accumulation is significantly reduced as compared to the estimation-scheme-only scenario.

The normalized tracking errors under the three scenarios for the validation trajectory #1 are presented in Figure 45 and similar results for validation trajectory #2 are presented in Figure 46. The percentage improvements of trajectory tracking errors for relevant metrics relative to the nominal conditions scenario for both validation trajectories are summarized in Table 5. Overall, these improvements vary from 3% to 93%. Significant improvements in the trajectory tracking errors can be observed when using the AIS approach as compared to using the estimation scheme alone. Promising performance is recorded for both the generation and validation data. The percentages change in errors with respect to the nominal errors are presented in Table 6-8. The data show that the AIS scenario has the capability to significantly reduce the errors that the estimation-scheme-only scenario produces. It is evident that the vertical channel appears to be particularly sensitive to accumulating errors. Depending on the initial altitude, the estimation-only scenario will frequently result in a crash, while the AIS scenario is capable of completing the mission.

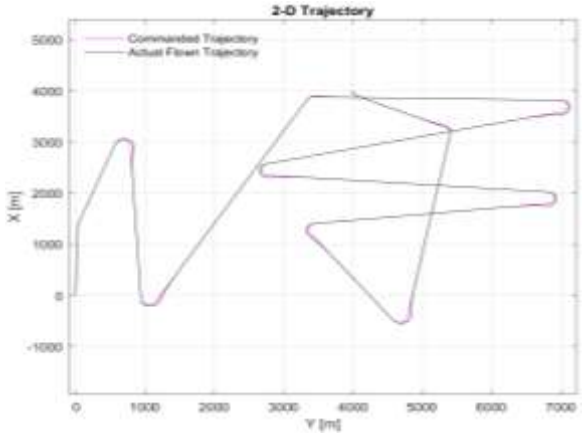

Figure 39. Nominal generation trajectory.

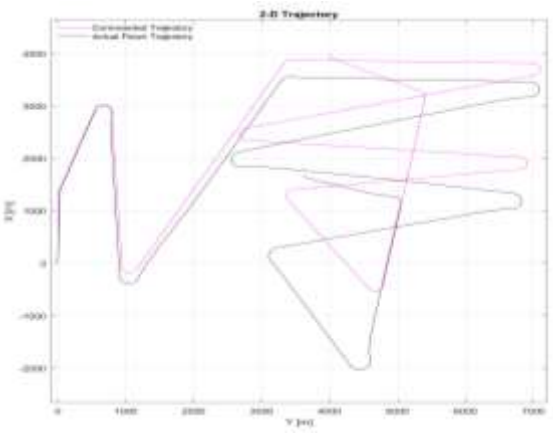

Figure 40. Estimation scheme generation trajectory.

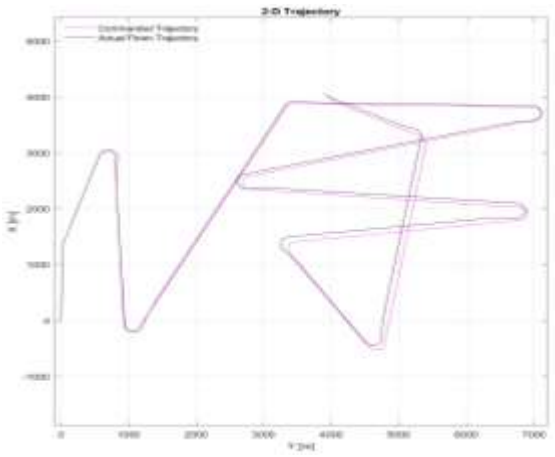

Figure 41. AIS generation trajectory.

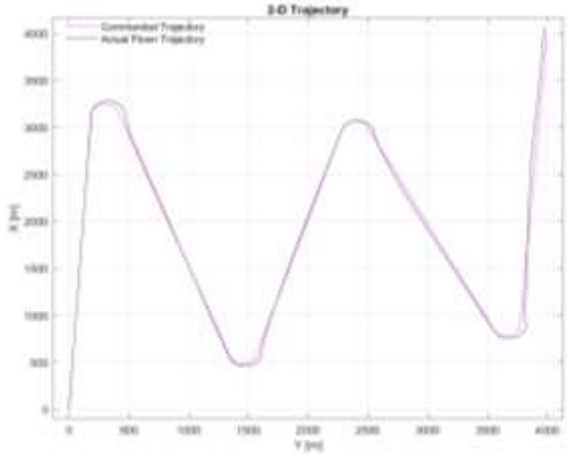

Figure 42. AIS first validation trajectory.

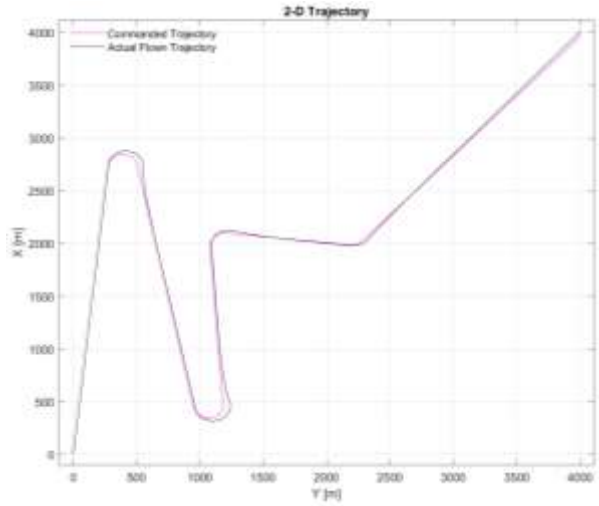

Figure 43. AIS second validation trajectory.

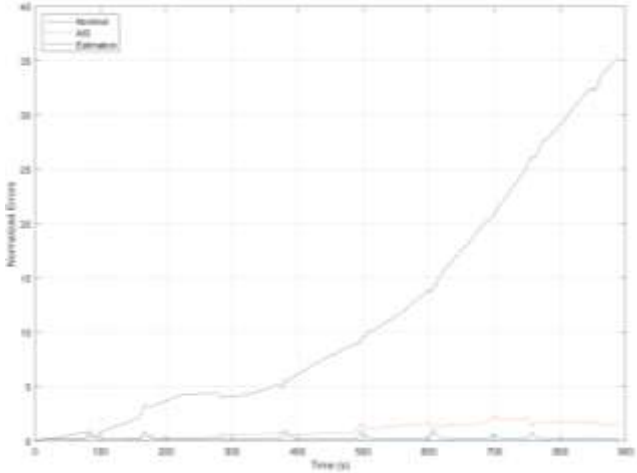

Figure 44. Normalized tracking errors of generation trajectory.

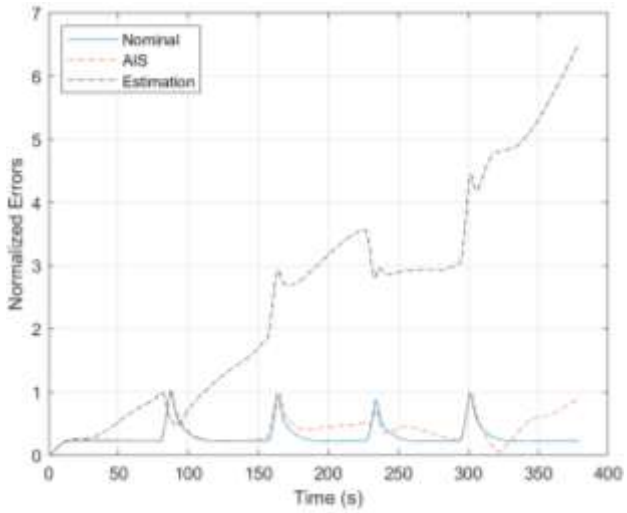

Figure 45. Normalized tracking errors of first validation trajectory.

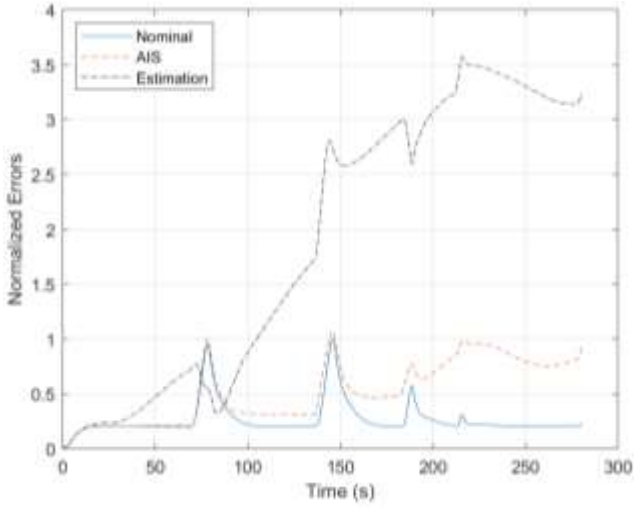

Figure 46. Normalized errors of second validation trajectory.

Table 5. The AIS percentage improvement.

Errors (%)	Generation	First Validation	Second Validation
Max XY Error [%]:	93	80	71
Max Z Error [%]:	94	90	10
Max XYZ Error [%]:	93	84	70
Mean XY Error [%]:	90	83	80
Mean Z Error [%]:	93	92	13
Mean XYZ Error [%]:	91	85	72
Standard Deviation XY Error [%]:	93	87	86
Standard Deviation Z Error [%]:	95	93	3
Standard Deviation XYZ Error [%]:	94	88	78

Table 6. Percentage change in errors for generation trajectory with respect to nominal errors.

Generation	AIS	IMU
Max XY Error [%]:	116	2790
Max Z Error [%]:	403	8572
Max XYZ Error [%]:	139	3450
Mean XY Error [%]:	210	3017
Mean Z Error [%]:	1908	29551
Mean XYZ Error [%]:	271	4220
Standard Deviation XY Error [%]:	271	5429
Standard Deviation Z Error [%]:	775	16139
Standard Deviation XYZ Error [%]:	342	7254

Table 7. Percentage change in errors for validation trajectory #1 with respect to nominal errors.

Errors	AIS	Estimation
Max XY Error [%]:	6	423
Max Z Error [%]:	58	1471
Max XYZ Error [%]:	5	549
Mean XY Error [%]:	30	676
Mean Z Error [%]:	166	3048
Mean XYZ Error [%]:	33	762
Standard Deviation XY Error [%]:	26	873
Standard Deviation Z Error [%]:	35	1790
Standard Deviation XYZ Error [%]:	26	975

Table 8. Percentage change in errors for validation trajectory #2 with respect to nominal errors.

Errors	AIS	Estimation
Max XY Error [%]:	2	256
Max Z Error [%]:	248	288
Max XYZ Error [%]:	7	257
Mean XY Error [%]:	35	589
Mean Z Error [%]:	924	1079
Mean XYZ Error [%]:	92	598
Standard Deviation XY Error [%]:	18	712
Standard Deviation Z Error [%]:	518	501
Standard Deviation XYZ Error [%]:	78	699

6.4. Effect of Navigation Accuracy on AIS

The accuracy of the position and velocity measurements used when generating the AIS will impact the quality of the AIS correction features. Different position measurement systems may be used to generate the AIS and “perfect” measurements can be obtained under certain scenarios. However, in this discussion, the position and velocity measurement system was referred to, generically, as GPS. A GPS that has low measurement errors generates more efficient memory cells that are more useful in the correction of position and velocity from the estimation scheme. The GPS for consumer purposes, such as smartphones has an accuracy of

about 4.9 m radius in open sky but this accuracy can be effected near obstacles, such as buildings and trees [85]. More expensive GPS with dual-frequency receivers and/or augmentation systems have high accuracy of within centimeters [86].

The errors between the commanded trajectory and actual trajectory in position and velocity for the generation trajectory #3 at normal conditions was measured with perfect accuracy algorithm and represented in Figure 47. A mathematical model was implemented to measure the position and velocity with very low errors, which are represented by the errors between the actual trajectory and the commanded trajectory at the nominal scenario, and these measurements are referred to as "perfect."

A Simulink GPS model was employed with a Precise Positioning Service (PPS) and had total contribution errors to the user equivalent range error GPS measurements of 1.4 m. More details about the PPS and GPS errors can be found in Kaplan and Hegarty [12]. The GPS model has the following parameters: 50 Hz position update rate, 50 Hz velocity update rate, 0.04-second time delay, 0.05-meter standard deviation of position error, 0.3 standard deviation ratio of velocity to position error, 2-meter maximum position bias, 1.5 maximum horizontal dilution of precision, and 2.5 maximum vertical dilution of precision. The GPS model possesses position and velocity measurement errors plotted in Figure 48. The AMC was created using perfect measurements from the mathematical model, then used to correct the position and velocity when the source of the perfect measurements was disabled.

The AMC was also created from the GPS model and used to correct the position and velocity when the GPS was not working. The simulation results show that the robustness of the autonomous flight control system is adequate and can tolerate a certain level of GPS errors with little trajectory tracking performance penalty. Table 9 illustrates that the measurement errors of the GPS model at normal conditions will slightly affect the tracking performance compared to the errors of the perfect measurements.

However, lower GPS performance produces larger compensation values of the correction features within the AIS. In turn, these values produce more substantial errors of the estimated corrected position and velocity and result in lower trajectory tracking performance. The AIS scenario still produces significant improvements as compared to the estimation-only approach. It

can be concluded that the GPS measurement errors have an important effect on reducing the percentage improvement of the AIS with respect to using the estimation scheme only for trajectory tracking. The results show 38% to 88% improvement for the GPS measurements and 66% to 95% improvement for perfect measurements. These results conclude that the errors of the GPS measurements have an impact on the AIS performance as shown in Figure 49, where the maximum normalized errors of the AIS built with GPS measurements at normal conditions were about twice the maximum errors of the AIS that was built with the perfect measurements at the same normal conditions. It should be emphasized again that the generation of the AIS does not necessarily require GPS to measure the vehicle position and velocity; this information can be obtained from alternative sources such as visual aids, and lidar and can be "perfect" under testing range conditions. However, if GPS was used to build the AIS, it must have high accuracy to ensure the quality of the AMC.

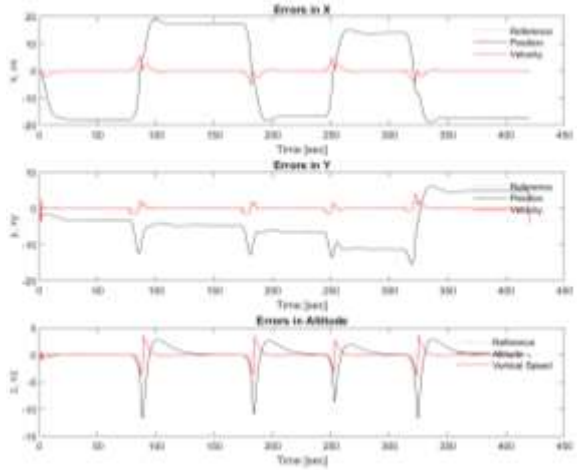

Figure 47. Perfect nominal errors.

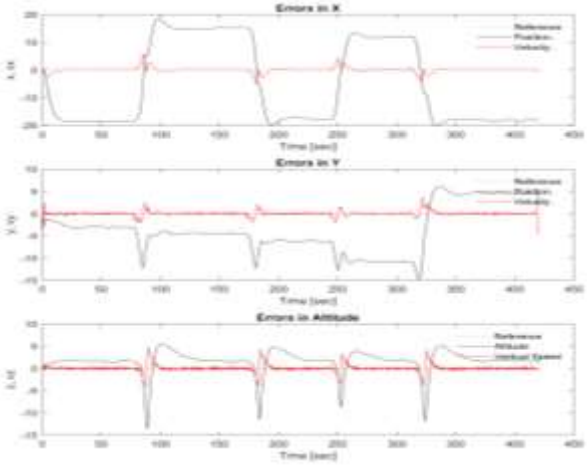

Figure 48. GPS nominal errors.

Table 9. Effect of GPS on errors.

Errors	Nominal Perfect [m]	Nominal GPS [m]	AIS Perfect [%]	AIS GPS [%]
Max XY Error:	20	21	95	82
Max Z Error:	12	12	66	39
Max XYZ Error:	20	21	85	69
Mean XY Error:	17	17	90	77
Mean Z Error:	1	2	66	38
Mean XYZ Error:	17	17	82	66
Standard Deviation XY Error:	3	3	99	92
Standard Deviation Z Error:	2	2	88	83
Standard Deviation XYZ Error:	2	3	90	84

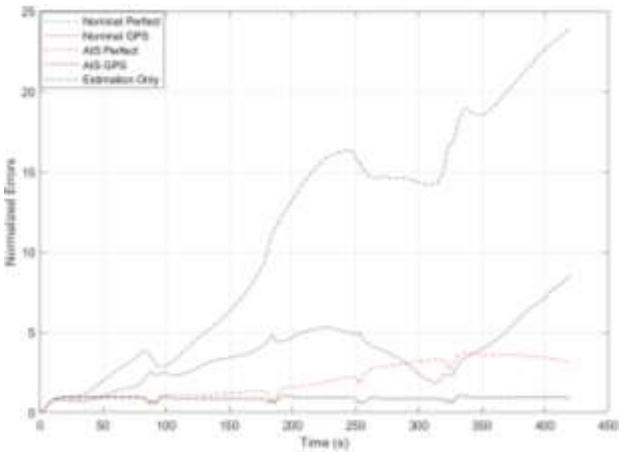

Figure 49. Effect of GPS on errors.

6.5. Path Planning Comparison

The differences between two types of pose-based UAV path generation methods, clothoid, and Dubins, have been investigated with the AIS approach using the generation trajectory #2. The Dubins path is a combination of circular arcs and straight-line segments, while the clothoid path arcs have a linearly variable curvature and are generated based on Fresnel integrals [8]. In [8], a comparison study between clothoid and Dubins demonstrated that clothoid facilitates better trajectory tracking than Dubins. In this research, AIS was built and tested with clothoid and then with Dubins. The normalized errors for the trajectory tracking were calculated and compared for clothoid and Dubins in Figures 50 and 51 respectively. The results show that clothoid has about 50% less maximum errors compared to Dubins. It is evident that building the AIS with Dubins and using it to correct the position and velocity of clothoid, would be the same as using the AIS with the validation trajectory and vice versa. Figure 52 shows the cross comparison using AMC built with Dubins and clothoid trajectories. Since nominal clothoid has less position and velocity errors than Dubins, the AMC built with clothoid has higher quality than the AMC built with Dubins. Thus, AIS performance is dependent on the quality of the AMC and navigation method used to build the AIS. Tables 10 and 11 show the AIS results for Dubins trajectory using AMC built with clothoid trajectory perform better than clothoid trajectory using AMC built with Dubins. Since Dubins appears to represent the weaker design option, the results in sections 6.2 and 6.3 were implemented using Dubins path planning to prove that AIS can work efficiently even with lower performance path planning.

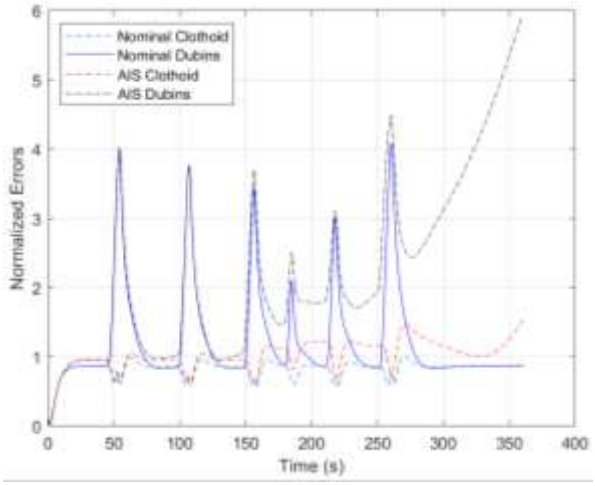

Figure 50. Dubins and clothoid trajectories AIS comparison without estimation.

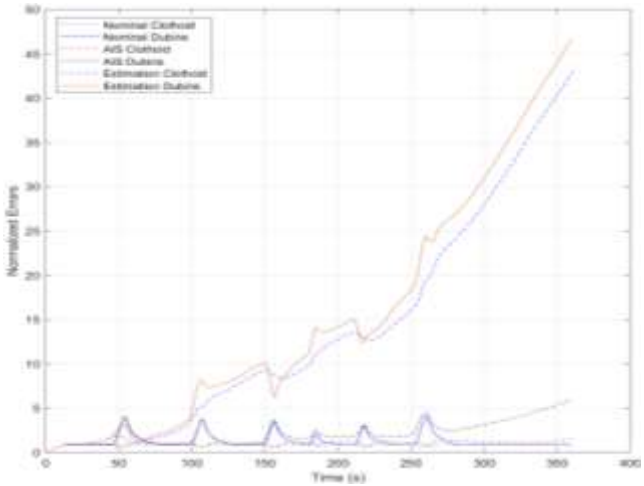

Figure 51. Dubins and clothoid trajectories AIS comparison with estimation.

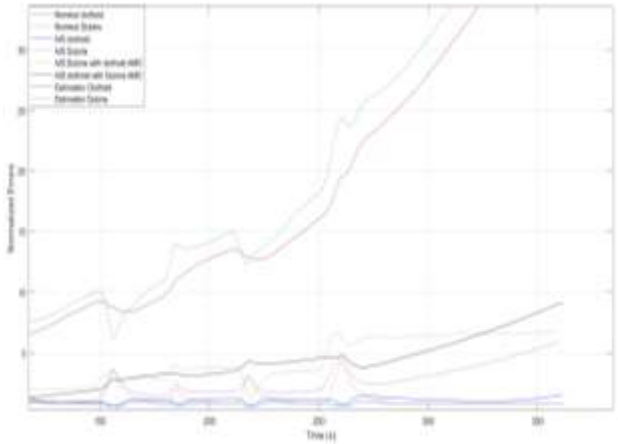

Figure 52. Cross comparison using AMC built with Dubins and clothoid trajectories.

Table 10. clothoid errors comparison

Errors with clothoid	Nominal	Estimation	AIS	AIS with Dubins AMC
Max XY Error [m]:	21	743	25	138
Max Z Error [m]:	14	510	26	57
Max XYZ Error [m]:	21	901	33	140
Mean XY Error [m]:	17	268	19	70
Mean Z Error [m]:	2	86	7	21
Mean XYZ Error [m]:	17	288	21	73
Standard Deviation XY Error [m]:	3	219	3	41
Standard Deviation Z Error [m]:	2	141	6	19
Standard Deviation XYZ Error [m]:	3	253	4	44

Table 11. Dubins errors comparison

Errors with Dubins	Nominal	Estimation	AIS	AIS with clothoid AMC
Max XY Error [m]:	83	795	82	55
Max Z Error [m]:	21	556	117	191
Max XYZ Error [m]:	85	970	123	191
Mean XY Error [m]:	25	293	33	23
Mean Z Error [m]:	4	100	27	60
Mean XYZ Error [m]:	26	317	46	69
Standard Deviation XY Error [m]:	15	236	14	15
Standard Deviation Z Error [m]:	4	154	29	49
Standard Deviation XYZ Error [m]:	15	275	27	45

6.6. IMU Grade Consideration

The accelerometer and gyroscope contain time-correlated random errors. The difference between the real value and the output is bias [87]. Bias errors are random constants that can be split into repeatability and stability bias depending on the behavior over time. The random accelerometer noise on the specific force measurements is integrated to produce a random-walk error on the inertial velocity solution. Moreover, gyro's random noise on the angular rate measurements is integrated to produce attitude random-walk error. The scale factor and cross-coupling errors are unitless and expressed in parts per million (ppm) [30]. The IMU types can be classified to several grades depending on their level of errors. Different grades of IMU are presented in Table 12 below [30, 75]:

Table 12. IMU grades.

Grade	Acc.Bias(m/s^2)	Gyro Bias(^{o}/ Hr)	Scale Factor
Marine	10^{-4}	0.001	<1/1000
Aviation	$3x10^{-4}$-10^{-3}	0.01	1/1000
Intermediate	10^{-3}-10^{-2}	0.1	1/100
Tactical	0.01-0.1	1-100	1
Consumer	>0.03	>100	50

All results presented so far were implemented using the IMU of aviation grade. However, the AIS can be built and used for tracking the commanded trajectory using lower grade IMU. Figure 53 shows the comparison between Intermediate grade IMU and the aviation IMU for AIS trajectory tracking. Table 13 shows a comparison of the performance metrics between these two grades of IMU with the AIS approach. The consideration of random error sources of the IMU was applied in this research. Usually, offline calibration can be used to overcome these error sources, but the on-run bias often varies with time gradually, making offline calibration less effective. An estimation of bias and gravity using a calibration technique and dynamic filtering was proposed by Batista and colleagues [88]. An intuitive approach to inertial sensor bias estimation was presented by Tereshkov [89] based on physical intuition and exploited a duality between gimbaled and strap-down inertial systems. The proposed method was tested on the land vehicle only. Two stages of estimations were presented; first, the means of feedback from an external aid to correct the inertial system attitude errors and steady-state feedback to rebalance uncompensated biases and estimate them. The second expresses the desired bias estimates as a feedback signal term. The information from such biased estimation algorithms could be introduced as features in the AMC that can handle the dynamic bias. Also, the time needed by such schemes can be limited such that the benefits of the AIS (increased “survivability”) would allow to obtain the current bias and compensate for it on time.

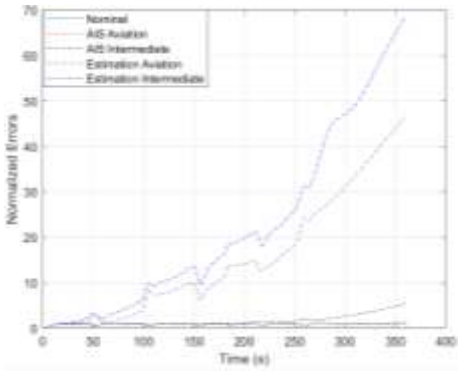

Figure 53. Effect of IMU grade on AIS

Table 13. Trajectory tracking errors of AIS for intermediate and aviation IMU grades.

	Intermediate	Aviation
Max XY Error [m]:	82	78
Max Z Error [m]:	205	18
Max XYZ Error [m]:	210	79
Mean XY Error [m]:	34	24
Mean Z Error [m]:	47	5
Mean XYZ Error [m]:	63	25
Standard Deviation XY Error [m]:	14	13
Standard Deviation Z Error [m]:	53	5
Standard Deviation XYZ Error [m]:	49	12

6.7. Affinity Methods Comparison

Two affinity methods based on the Manhattan and Euclidean distances were compared using the generation trajectory #4. The difference between using Manhattan and Euclidean affinity methods with AIS is shown in Figure 54. Both methods showed almost the same normalized errors with respect to the nominal scenario. The main performance evaluation metrics for nominal scenario and the two compared affinity methods with the AIS scenario are shown in Table 14. The Euclidean distance method produces slightly fewer errors than the Manhattan distance method, varying between 1 and 3m over the set of considered metrics. The errors in both approaches are considerably less than those occurring when only using the estimation scheme and using either one does not change the quality of the AIS results. For all other results presented in this research, the Manhattan affinity was used to implement the AIS.

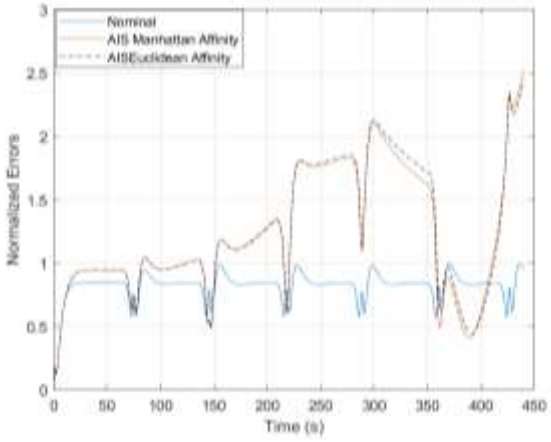

Figure 54. Normalized tracking errors for different affinity methods.

Table 14. Errors comparison of affinity methods with AIS.

Errors	Nominal	Euclidean	Manhattan
Max XY Error [m]:	21	36	36
Max Z Error [m]:	15	392	395
Max XYZ Error [m]:	21	394	397
Mean XY Error [m]:	17	21	20
Mean Z Error [m]:	2	73	73
Mean XYZ Error [m]:	18	80	81
Standard Deviation XY Error [m]:	3	8	9
Standard Deviation Z Error [m]:	3	101	102
Standard Deviation XYZ Error [m]:	2	97	98

6.8. UAV Features Analysis

Each UAV feature has its own impact on the accuracy of the matching algorithm in finding the index of the best antibody from AMC. A methodology has been developed to determine the weight of the impact of each UAV feature on the matching algorithm accuracy. The determined index (i) will be used to select the UAV features from AMC ($\varphi_{s(j)}^{(i)}$) with the same index row number. Then, by subtracting each selected feature from its counterpart from the IMU features ($\varphi_{s(j)}^{(\propto)}$), the differences with values close to zero will represent the best matching between them. A matching performance index M_j for each vehicle feature j is defined in terms of the average μ of the absolute values of the calculated differences over the full flight duration. This performance index can be used to assess the capability of the vehicle features to relate to the corrections. It can represent a metric for selecting the most relevant vehicle feature with potential for increasing AIS performance.

$$M_j = \mu\left\|\varphi_s^{(\propto)} - \varphi_{s(j)}^{(i)}\right\| \approx 0 \tag{79}$$

where μ is the statistical mean, i is the index number, and j is the sequence number of the UAV feature, and $\propto$ is the current time step.

The block diagram in Figure 55 demonstrates the effect of each UAV feature on the matching algorithm of generation trajectory #1. The results of the block diagram are shown in Figures 56 and 57, respectively. From these figures, it can be concluded that the angular rates show the best matching capability because of their minimum matching performance index values. The acceleration components show the highest mismatching values.

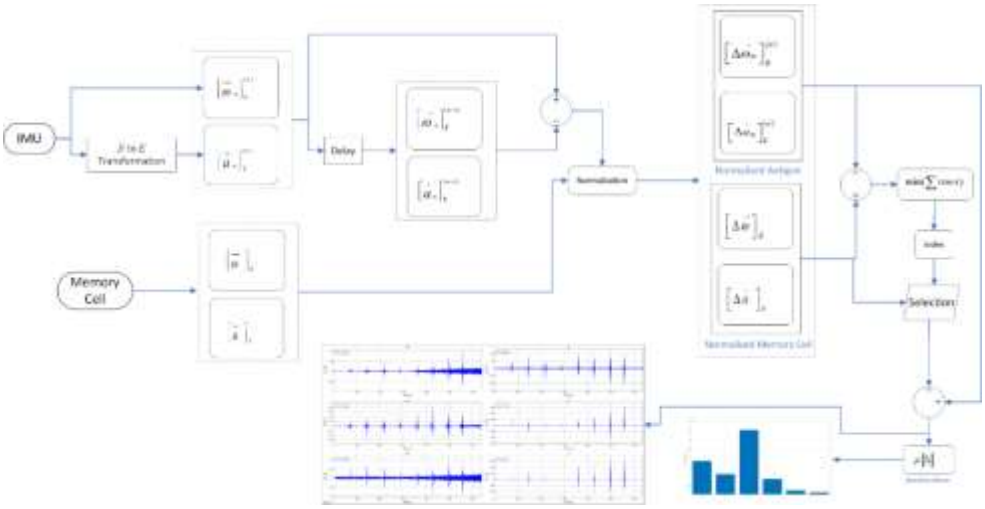

Figure 55. Determination of UAV features effect on matching algorithm.

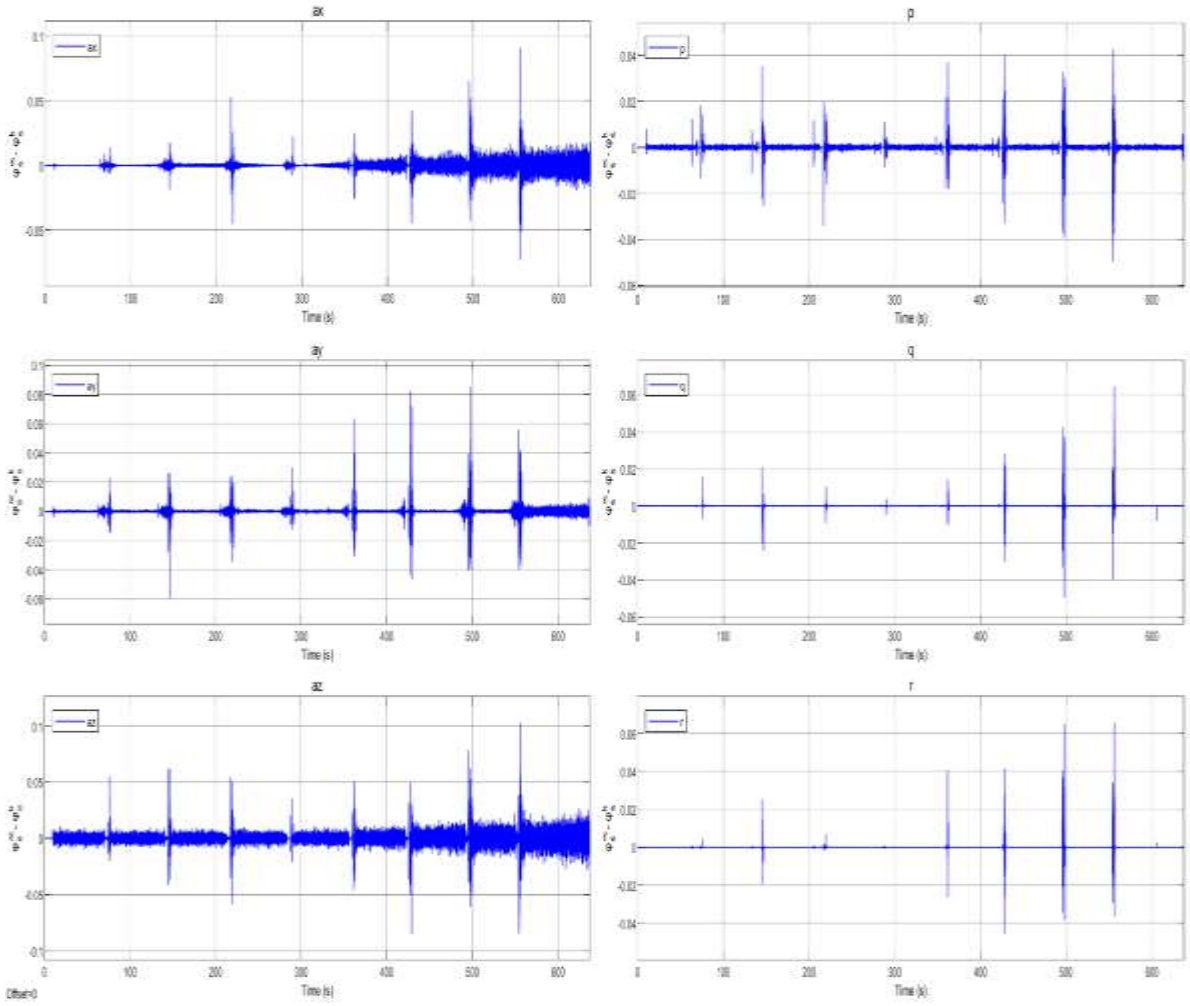

Figure 56. UAV features matching accuracy.

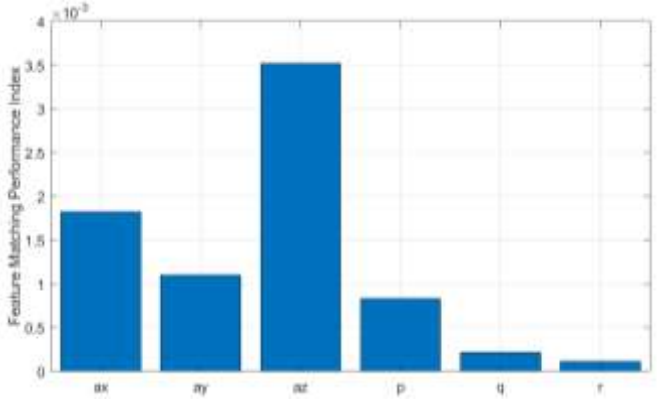

Figure 57. Weighted mismatching of UAV features.

To illustrate the investigated method, the vehicle feature set is extended with the angular acceleration components to generate the AMC, each cell M_j denoted as:

$$M_j = \left[\Delta a_{xj} \;\; \Delta a_{yj} \;\; \Delta a_{zj} \;\; \Delta p_j \;\; \Delta q_j \;\; \Delta r_j \;\; \Delta \dot{p}_j \;\; \Delta \dot{q}_j \;\; \Delta \dot{r}_j \;\vdots\; \varphi_{cj}\right] \tag{80}$$

where:

$$\dot{p} = \frac{dp}{dt} \tag{81}$$

$$\dot{q} = \frac{dq}{dt} \tag{82}$$

$$\dot{r} = \frac{dr}{dt} \tag{83}$$

The angular acceleration was used as a UAV feature to investigate its effect on the matching algorithm. The results show that using the angular acceleration reduces the accuracy of the matching algorithm and changes the effects of other features on the matching accuracy. As shown in Figure 58, the matching algorithm indices correlate to the worse performance of the AIS when the new features are added.

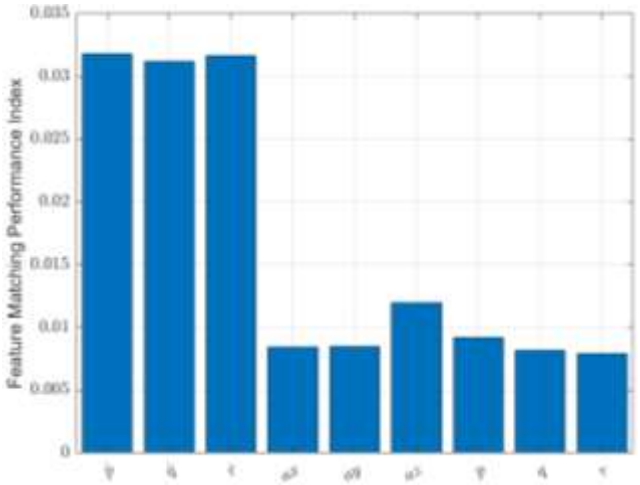

Figure 58. Effect of including the angular acceleration on the matching accuracy.

In general, using the angular accelerations $\vec{\omega}$ as UAV features reduces the percentage improvement of the AIS validation trajectory results and does not add any significant improvement to the generation trajectory (see Tables 15 and 16); therefore, it is not efficient to include it with the UAV features set as shown in Figure 59; thus, was not considered in this research.

Table 15. Effect of angular acceleration and angular rate on AIS generation trajectory.

Generation Trajectory	% Improvement		
	AIS with $\vec{\omega}$	AIS with $\vec{\omega}$	AIS with $\vec{\omega}$ & $\vec{\omega}$
Max XY Error	96	93	96
Max Z Error	72	94	62
Max XYZ Error	83	93	77
Mean XY Error	96	90	96
Mean Z Error	82	93	74
Mean XYZ Error	89	91	84
Standard Deviation XY Error	97	93	97
Standard Deviation Z Error	79	95	66
Standard Deviation XYZ Error	87	94	80

Table 16. Effect of angular acceleration and angular rate on AIS validation trajectory.

Validation Trajectory	% Improvement		
	AIS with $\vec{\omega}$	AIS with $\vec{\omega}$	AIS with $\vec{\omega}$ & $\vec{\omega}$
Max XY Error	71	80	72
Max Z Error	80	90	78
Max XYZ Error	74	84	74
Mean XY Error	71	83	73
Mean Z Error	68	92	68
Mean XYZ Error	70	85	72
Standard Deviation XY Error	71	87	72
Standard Deviation Z Error	78	93	77
Standard Deviation XYZ Error	74	88	74

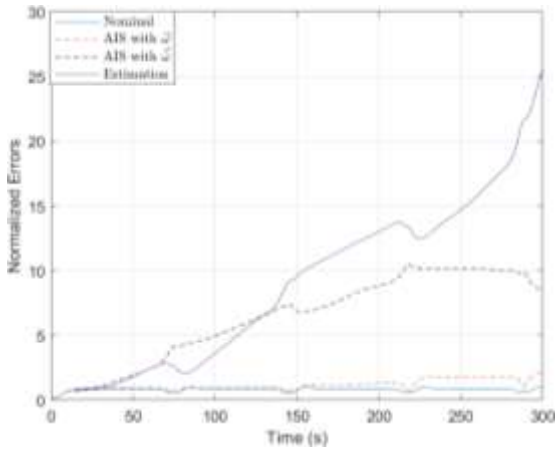

Figure 59. Effect of $\vec{\omega}$ features on the AIS performance.

6.9. Augmentation of Position and Velocity Vertical Components

Altitude and vertical speed can be obtained using a variety of techniques such as GNSS, pressure altimeter, laser, radar, lidar, and visual sensors. Some of these sensor systems are affordable and can be installed on-board independently with respect to external infrastructure. The indication of height in terms of pressure altitude and vertical speed in term of the rate of change in the pressure altitude can be measured using the static pressure that actuates the altimeter and the vertical speed indicator [90]. Measuring the vertical components (z, vz) is less complex than measuring the horizontal components (x, y, vx, vy) and is easier to accomplish with low cost and high accuracy. If the UAV control laws are provided with outputs of adequate accuracy from altitude sensor or multi-sensors that work independently of the GNSS and other external sources, then the AIS can exclude the vertical components from the correction features, and improvements in overall performance can potentially be achieved. Figure 60 shows the case when the AIS is not used to estimate and correct the vertical components (z, vz); instead these components are compensated from another on-board sensor. Employing additional on-board sensor output for vertical components with good accuracy is expected to improve the trajectory tracking performance of the AIS-based compensation. Tables 17 and 18 show the performance metrics comparisons and improvements for generation trajectory #1 and validation trajectory #1 using clothoid path planning. Table 17 shows the results when AIS compensation was used on all three channels, and Table 18 shows the results when on-board additional sensors provide the vertical components without using external information sources.

These tables show that, when using additional vertical measurements, the performance metrics will exhibit a significant improvement, with errors reductions between 2% to 70 %. Figures 61 and 62 show the improvement to the normalized errors to the trajectory tracking when using AIS only or AIS and vertical components measurements with generation trajectory #1. Similar results for the validation trajectory #1 are shown in Figures 63 and 64. From all these tests, it can be concluded that accurate on-board sensors for altitude have the potential to augment the AIS and improve tracking of the commanded trajectory. Indeed, the accuracy of the vertical component measurements affects the results of the trajectory tracking and AIS performance. The accuracy can vary depending on the type of the used sensor and the method of measurement. However, to improve the AIS performance, vertical components measurement errors should not significantly exceed the GNSS errors by more than 20%.

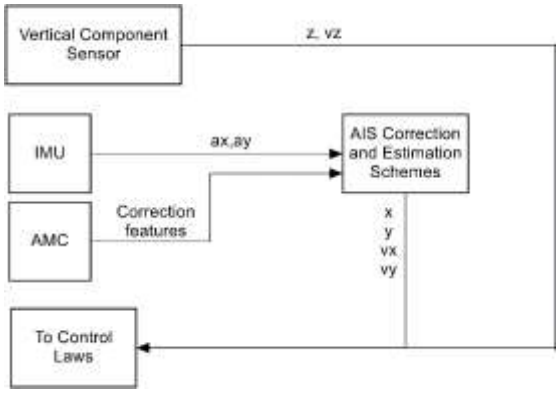

Figure 60. Vertical components compensation.

Table 17. Performance metrics comparison without using vertical component sensor augmentation.

	Generation				Validation			
Errors	Nominal [m]	AIS [m]	Estimation [m]	% Improvement	Nominal [m]	AIS [m]	Estimation [m]	% Improvement
Max XY Error:	20	87	2295	96	20	61	467	87
Max Z Error:	12	886	1787	50	12	86	279	69
Max XYZ Error:	20	890	2906	69	20	97	544	82
Mean XY Error:	17	27	693	96	17	31	199	84
Mean Z Error:	1	194	641	70	1	28	85	67
Mean XYZ Error:	17	200	957	79	17	46	219	79
Standard Deviation XY Error:	3	18	617	97	3	16	136	88
Standard Deviation Z Error:	2	251	603	58	2	30	68	55
Standard Deviation XYZ Error:	2	248	847	71	2	29	148	80

Table 18. Performance metric comparison with vertical component sensor augmentation.

	Generation				Validation			
Errors	Nominal [m]	AIS [m]	Estimation [m]	% Improvement	Nominal [m].	AIS [m]	Estimation [m]	% Improvement
Max XY Error:	20.417	57	2284	98	20	63	490	87
Max Z Error:	14	13	20	33	12	13	13	4
Max XYZ Error:	21	57	2284	98	20	63	490	87
Mean XY Error:	17	23	682	97	17	29	205	86
Mean Z Error:	1	1	1	0	1	1	1	0
Mean XYZ Error:	18	23	682	97	17	29	205	86
Standard Deviation XY Error:	2	11	605	98	3	14	142	90
Standard Deviation Z Error:	2	2	2	0	2	2	2	0
Standard Deviation XYZ Error:	2	11	605	98	2	14	142	90

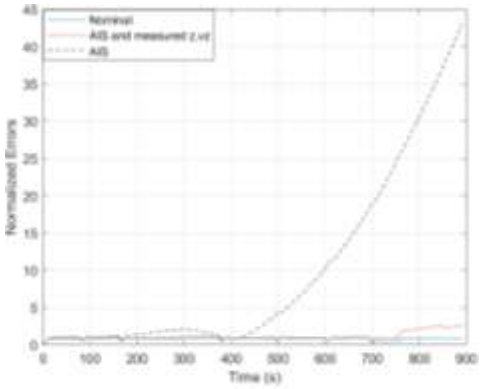

Figure 61. Generation trajectory #1 normalized errors.

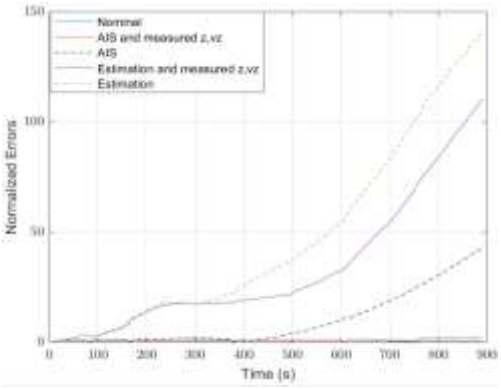

Figure 62. Generation trajectory #1 normalized errors with estimation.

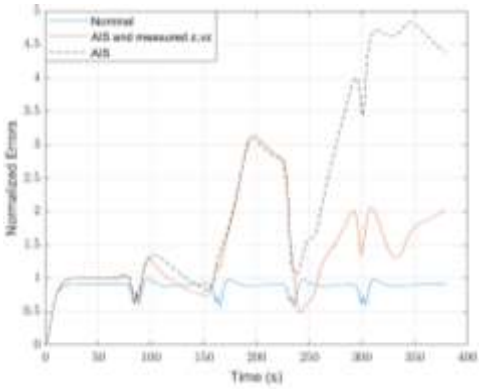

Figure 63. Validation trajectory #1 normalized errors.

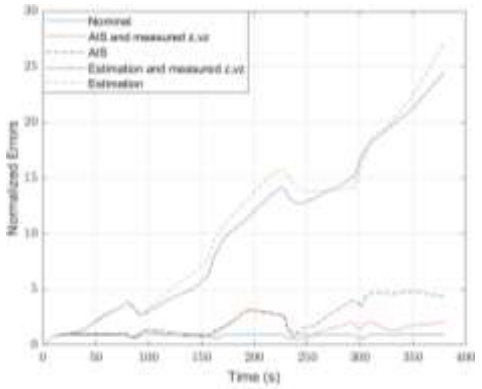

Figure 64. Validation trajectory #1 normalized errors with estimation.

7. Conclusion

A general framework was proposed, implemented, and successfully tested through simulation for the correction of position and velocity estimates based only on the inertial sensor output. AIS-corrected estimations of the velocity and position are the values obtained from the estimation scheme plus the corrections extracted from the AIS. The AIS has been shown to possess good classification and memory capabilities that can be used successfully for autonomous flight control purposes. The proposed framework employs data-driven methods. The AIS is structured as a set of artificial memory cells that mimic specialized immunity cells and may be generated based on actual system operation, testing range data, and even simulation.

The primary objective of using the AIS approach is to achieve adequate navigation that is trajectory independent and covers all possible dynamic configurations. To achieve this higher level of generality, one-time-step "deltas" were used in defining the UAV features as a representation of the system dynamic configuration. Simulation results have demonstrated the promising potential of this solution for significantly reducing integral error accumulation.
The immunity-based framework for autonomous flight in a GPS-denied environment was implemented in the WVU UAS simulation environment. The artificial immune system was built with different trajectories and tested using generation and validation trajectories. Tracking the AIS generation trajectory demonstrated effective performance during the GPS-denied operation. The AIS paradigm shows potential for capturing estimation errors and providing corrections to position and velocity estimation algorithms. Promising improvement of trajectory tracking performance has also been obtained for validation commanded trajectories other than the AIS generation trajectories. The AIS paradigm exhibited better results with the generation trajectory as compared to the validation trajectory, meaning that it will be a perfect solution for UAVs that use the same trajectory in repetitive missions such as agricultural usage and cargo delivery.

The performance of the proposed methodology can be improved further by using more generation data and more vehicle features with better selectivity including dynamic features. A method for assessing the potential of new vehicle features for improving AIS potential has been developed and successfully tested in this research effort. The proposed data-driven methodology did not exhibit the drawbacks of existing approaches for handling GNSS-denied situations. The

proposed methodology does not require external infrastructure or information of opportunity for system actual operation. The method requires the availability of accurate vehicle position and velocity for AIS generation. However, this information can be obtained from nominally working GNSS at different locations or alternative sources including well-controlled laboratory settings.

A significant step was achieved in this research towards developing a comprehensive and integrated solution for monitoring and controlling aerospace systems including navigation and trajectory tracking. A novel framework was developed and tested extensively for UAV trajectory tracking based on principles of artificial immune systems with a newly proposed structure referred to as artificial memory cells. The effectiveness and performance of the proposed immunity-based framework was analyzed and demonstrated to represent a promising solution. The effect of path the planning algorithm for the commanded trajectory, model of the affinity method, class of the selected sensors, selected features for the matching algorithm, and scenario when an external source of the vertical components measurements is available on the proposed method were investigated opening promising avenues for future research. Two approaches for the proposed method were developed: correction of the estimation scheme input and correction of the estimation scheme output. Both approaches produced considerable improvement to the accumulating errors of the inertial measurement sensor; however, the correction of the estimation scheme input had a higher percentage improvement than the correction of the estimation scheme output and is less computationally expensive.

The results of the AIS depend on the quality of the AMC; thus, the quality of the path planning and navigation source during the generation of the AIS, the grade of the inertial measurement sensor, and the number of the UAV features that have better selectivity. The AIS paradigm can work jointly with augmentation of position and velocity vertical components to exhibit adequate improvement to the horizontal components.

8. References

[1] The United States Department of Defence, *Unmanned systems integrated roadmap: FY2013-2038*, Washington, DC, USA, 2013. [Online]. Available: https://apps.dtic.mil/dtic/tr/fulltext/u2/a592015.pdf.

[2] R. Blockley, E. Atkins, A. Ollero, W. Shyy, and A. Tsourdos, *Unmanned Aircraft Systems*. Chichester, UK: Wiley, 2017.

[3] R. Bartsch, J. Coyne, and K. Gray, *Drones in Society: Exploring the strange new world of unmanned aircraft*. New York, NY: Taylor & Francis, 2016.

[4] G. Conte and P. Doherty, "An integrated UAV navigation system based on aerial image matching," in *2008 IEEE Aerospace Conference*, 2008: IEEE, pp.1-10. [Online]. Available: doi: 10.1109/AERO.2008.4526556.

[5] M. Pitchford. (2018, 08/07/2018). *What's needed to ensure safety and security in UAV software* [Online]. Available: http://mil-embedded.com/articles/whats-needed-ensure-safety-security-uav-software/.

[6] F. Dovis, *GNSS interference threats and countermeasures*. Norwood, MA: Artech House, 2015.

[7] B. K. Wilburn, M. G. Perhinschi, H. Moncayo, O. Karas, and J. Wilburn, "Unmanned aerial vehicle trajectory tracking algorithm comparison," *International Journal of Intelligent Unmanned Systems,* vol.1, no.3, pp.276-302, 2013. [Online]. Available: https://doi.org/10.1108/IJIUS-05-2013-0018.

[8] M. Al Nuaimi, "Analysis and comparison of clothoid and Dubins algorithms for UAV trajectory generation," M.S. Thesis, West Virginia University, Morgantown, WV, 2014.

[9] M. Sadraey, "Unmanned aircraft design: A review of fundamentals," *Synthesis Lectures on Mechanical Engineering,* vol.1, no.2, pp.i-193, 2017. [Online]. Available: doi: 10.2200/S00789ED1V01Y201707MEC004.

[10] J. J. Spilker Jr, P. Axelrad, B. W. Parkinson, and P. Enge, *Global positioning system: Theory and applications, Volume I*. Washington, D.C.: American Institute of Aeronautics and Astronautics, Inc., 1996.

[11] C. Jeffrey, "An introduction to GNSS: GPS, GLONASS, Galileo and other global navigation satellite systems," *NovAtel Inc,* 2010.

[12] E. Kaplan and C. Hegarty, *Understanding GPS: Principles and applications*, 2nd ed. Norwood, MA: Artech House, 2005.

[13] T. Fossen, K. Y. Pettersen, and H. Nijmeijer, *Sensing and Control for Autonomous Vehicles*. Cham, Switzerland: Springer, 2017.

[14] D. P. Shepard, J. A. Bhatti, T. E. Humphreys, and A. A. Fansler, "Evaluation of smart grid and civilian UAV vulnerability to GPS spoofing attacks," in *Proceedings of the ION GNSS Meeting*, 2012, vol.3, pp.3591-3605. [Online]. Available: https://www.xdrones.es/wp-content/uploads/2016/07/PMUAndUAVSpoofingION2012.pdf.

[15] S. Zhao, C. Huang, X. Qi, and M. Lu, "Application of BeiDou navigation satellite system on attitude determination for Chinese Space Station," in *China Satellite Navigation Conference (CSNC) 2015 Proceedings: Volume I*, J. Sun, J. Liu, S. Fan, and X. Lu, Eds. Heidelberg, Berlin: Springer, 2015, Ch.2, pp.13-25.

[16] F. G. Toro and A. Tsourdos, *UAV sensors for environmental monitoring*. MDPI, 2018.

[17] R. M. Watson and J. N. Gross, "Robust navigation in GNSS degraded environment using graph optimization," *arXiv preprint arXiv:1806.08899,* 2018. [Online]. Available: https://arxiv.org/pdf/1806.08899.pdf.

[18] F. J. Perez-Grau, R. Ragel, F. Caballero, A. Viguria, and A. J. J. o. F. R. Ollero, "An architecture for robust UAV navigation in GPS-denied areas," vol.35, no.1, pp.121-145, 2018. [Online]. Available: https://doi.org/10.1002/rob.21757.

[19] V. O. Sivaneri and J. N. Gross, "UGV-to-UAV cooperative ranging for robust navigation in GNSS-challenged environments," *Aerospace Science and Technology,* vol.71, pp.245-255, 2017. [Online]. Available: https://doi.org/10.1016/j.ast.2017.09.024.

[20] V. O. Sivaneri and J. N. Gross, "Cooperative navigation between a ground vehicle and an unmanned aerial vehicle in GNSS-challenged environments," in *Proceedings of the 29th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2016), ION*, 2016, pp.1512-1521. [Online]. Available: https://doi.org/10.33012/2016.14623.

[21] G. Balamurugan, J. Valarmathi, and V. Naidu, "Survey on UAV navigation in GPS denied environments," in *Signal Processing, Communication, Power and Embedded System (SCOPES), 2016 International Conference on*, 2016: IEEE, pp.198-204. [Online]. Available: doi: 10.1109/SCOPES.2016.7955787.

[22] M. Perhinschi, M. Napolitano, and S. Tamayo, "Integrated simulation environment for unmanned autonomous systems: Towards a conceptual framework," *Modelling and Simulation in Engineering,* vol.2010, 2010, Art no. 2. [Online]. Available: 10.1155/2010/736201.

[23] D. Dasgupta, *Artificial immune systems and their applications*. Berlin; London: Springer, 1999.

[24] E. Benjamini, G. Sunshine, and R. Coico, *Immunology: a short course*, 4th ed. / Eli Benjamini, Richard Coico, Geoffrey Sunshine. ed. New York; Chichester: Wiley-Liss, 2000.

[25] U. Aickelin and D. Dasgupta, "Artificial immune systems," in *Search methodologies*: Springer, 2005, pp.375-399. [Online]. Available: https://doi.org/10.1007/0-387-28356-0_13.

[26] R. J. Doyle, "Autonomy needs and trends in deep space exploration," California Institute of Technology Pasadena Jet Propulsion Lab, 2003, [Online]. Available: https://apps.dtic.mil/dtic/tr/fulltext/u2/a485027.pdf.

[27] National Research Council, *Autonomy research for civil aviation: Toward a new era of flight*: National Academies Press, 2014. [Online]. Available: https://www.nap.edu/resource/18815/deps_144680.pdf.

[28] R. Austin, *Unmanned aircraft systems: UAVS design, development and deployment, Vol.54*. Hoboken, N.J: John Wiley & Sons, 2011.

[29] K. Mohta *et al.*, "Fast, autonomous flight in GPS-denied and cluttered environments," vol.35, no.1, pp.101-120, 2018. [Online]. Available: https://doi.org/10.1002/rob.21774.

[30] P. D. Groves, *Principles of GNSS, inertial, and multisensor integrated navigation systems*. Artech House, 2013.

[31] Q.-L. Zhou, Y. Zhang, Y.-H. Qu, and C.-A. Rabbath, "Dead reckoning and Kalman filter design for trajectory tracking of a quadrotor UAV," in *Proceedings of 2010 IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications*, 2010: IEEE, pp.119-124. [Online]. Available: doi: 10.1109/MESA.2010.5552088.

[32] O. Speck, D. Speck, R. Horn, J. Gantner, and K. P. Sedlbauer, "Biomimetic bio-inspired biomorph sustainable? An attempt to classify and clarify biology-derived technical developments," *Bioinspiration & Biomimetics,* vol.12, no.1, p.011004, 2017. [Online]. Available: doi: 10.1088/1748-3190/12/1/011004.

[33] X.-S. Yang, *Nature-inspired optimization algorithms*. London, UK: Elsevier, 2014.

[34] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. Cambridge, MA: MIT Press, 1992.

[35] I. K. Nikolos, K. P. Valavanis, N. C. Tsourveloudis, and A. N. Kostaras, "Evolutionary algorithm based offline/online path planner for UAV navigation," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics),* vol.33, no.6, pp.898-912, 2003.

[36] G. E. U. Faelden, J. M. Z. Maningo, R. C. S. Nakano, A. A. Bandala, and E. P. Dadios, "Blind localization method for quadrotor-unmanned aerial vehicle (QUAV) utilizing genetic algortihm," in *Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), 2014 International Conference on*, 2014: IEEE, pp.1-5. [Online]. Available: doi: 10.1109/HNICEM.2014.7016214.

[37] B. K. Wilburn, M. G. Perhinschi, and J. N. Wilburn, "A modified genetic algorithm for UAV trajectory tracking control laws optimization," *International Journal of Intelligent Unmanned Systems,* vol.2, no.2, pp.58-90, 2014. [Online]. Available: https://doi.org/10.1108/IJIUS-03-2014-0002.

[38] M. Dorigo, G. D. Caro, and L. M. Gambardella, "Ant algorithms for discrete optimization," *Artificial life,* vol.5, no.2, pp.137-172, 1999. [Online]. Available: doi: 10.1162/106454699568728.

[39] P. Graham and A. Philippides, "Vision for navigation: What can we learn from ants?," *Arthropod Structure & Development,* vol.46, no.5, pp.718-722, 2017. [Online]. Available: doi: 10.1016/j.asd.2017.07.001.

[40] M. Dorigo, V. Maniezzo, A. Colorni, and V. Maniezzo, "Positive feedback as a search strategy," 1991.

[41] G. Ma, H. Duan, and S. Liu, "Improved ant colony algorithm for global optimal trajectory planning of UAV under complex environment," *International Journal on Computational Science & Application,* vol.4, no.3, pp.57-68, 2007. [Online]. Available: http://www.tmrfindia.org/ijcsa/V4I35.pdf.

[42] S. Amari, *The handbook of brain theory and neural networks*. Cambridge, MA: MIT Press, 2003.

[43] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of Mathematical Biophysics,* vol.5, no.4, pp.115-133, 1943. [Online]. Available: https://doi.org/10.1007/BF02478259.

[44] X. Guan and C. Cai, "A new integrated navigation system for the indoor unmanned aerial vehicles (UAVs) based on the neural network predictive compensation," in *2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, 2018: IEEE, pp.575-580. [Online]. Available: doi: 10.1109/YAC.2018.8406440.

[45] M. Perhinschi, H. Moncayo, B. Wilburn, J. Wilburn, O. Karas, and A. Bartlett, "Neurally-augmented immunity-based detection and identification of aircraft sub-system failures," *The Aeronautical Journal,* vol.118, no.1205, pp.775-796, 2014. [Online]. Available: doi: 10.1017/s0001924000009532.

[46] L. A. Zadeh, "Fuzzy sets," *Information and Control,* vol.8, no.3, pp.338-353, 1965. [Online]. Available: https://doi.org/10.1016/S0019-9958(65)90241-X.

[47] L. A. Zadeh, "Fuzzy sets and information granurity," in *Advances in fuzzy set theory and applications*, M. M. Gupta, R. K. Ragade, and R. R. Yager, Eds. Amsterdam: North-Holland Publishing Company, 1979, pp.3-18. [Online]. Available: https://pdfs.semanticscholar.org/4bd8/911be1f5a880ab48fcf91fd19709ae86c8ef.pdf.

[48] L. A. Zadeh, "Possibility theory and soft data analysis," *Mathematical frontiers of the social and policy sciences,* pp.69-129, 1981. [Online]. Available: https://doi.org/10.1142/9789814261302_0025.

[49] M. Perhinschi, "Parameter optimization via genetic algorithm of fuzzy controller for autonomous air vehicle," in *Proceedings of the AIAAGuidance, Navigation, and Control Conference and Exhibit*, Portland OR, USA, August 1999, pp.790-797. [Online]. Available: doi: 10.2514/6.1999-4084.

[50] C. Sabo and K. Cohen, "Fuzzy logic unmanned air vehicle motion planning," *Advances in Fuzzy Systems,* vol.2012, p.14, 2012, Art no. 989051. [Online]. Available: http://dx.doi.org/10.1155/2012/989051.

[51] Z. Sun, T. Dong, X. Liao, R. Zhang, and D. Y. Song, "Fuzzy logic for flight control II: fuzzy logic approach to path tracking and obstacles avoidance of UAVs," in *Advanced Fuzzy Logic Technologies in Industrial Applications*, Y. Bai, H. Zhuang, and D. Wang, Eds. London: Springer London, 2006, pp.223-235. [Online]. Available: https://doi.org/10.1007/978-1-84628-469-4_15.

[52] J. Wilburn, J. Cole, M. Perhinschi, and B. Wilburn, "Comparison of a fuzzy logic controller to a potential field controller for real-time UAV navigation," in *AIAA Guidance, Navigation, and Control Conference*, 2012, p.4907. [Online]. Available: doi: 10.2514/6.2012-4907.

[53] A. Abdul Rasheed and M. Mohamed Sathik, "Moblog-Based Social Networks," in *Social Networks: A Framework of Computational Intelligence*, W. Pedrycz and S.-M. Chen, Eds. Cham: Springer International Publishing, 2014, pp.75-97. [Online]. Available: https://doi.org/10.1007/978-3-319-02993-1_5.

[54] J. Kennedy, "Particle swarm optimization," in *Encyclopedia of machine learning*: Springer, 2011, pp.760-766.

[55] S. Ahmadzadeh and M. Ghanavati, "Navigation of mobile robot using the PSO particle swarm optimization," *Journal of Academic and Applied Studies,* vol.2, no.1, pp.32-38, 2012.

[56] D. Dasgupta, "Advances in artificial immune systems," *IEEE Computational Intelligence Magazine,* vol.1, no.4, pp.40-49, 2006. [Online]. Available: doi: 10.1109/MCI.2006.329705.

[57] M. G. Perhinschi, H. Moncayo, and J. Davis, "Integrated framework for artificial immunity-based aircraft failure detection, identification, and evaluation," *Journal of Aircraft,* vol.47, no.6, pp.1847-1859, 2010. [Online]. Available: https://arc.aiaa.org/doi/10.2514/1.45718.

[58] M. G. Perhinschi, H. Moncayo, and D. Al Azzawi, "Integrated immunity-based framework for aircraft abnormal conditions management," *Journal of Aircraft,* vol.51, no.6, pp.1726-1739, 2014. [Online]. Available: https://doi.org/10.2514/1.C032381.

[59] M. Perhinschi and H. Moncayo, "Artificial immune system for comprehensive and integrated aircraft abnormal conditions management," in *Advances in Computational Intelligence and Autonomy for Aerospace Systems AIAA Series Progress in Aeronautics and Astronautics*, J. Valasek, Ed., August 2018, pp.147-218.

[60] A. Togayev, M. Perhinschi, H. Moncayo, D. Al Azzawi, and A. Perez, "Immunity-based accommodation of aircraft subsystem failures," *Aircraft Engineering and Aerospace Technology,* vol.89, no.1, pp.164-175, 2017. [Online]. Available: https://doi.org/10.1108/AEAT-08-2014-0124.

[61] S. Sanchez, M. Perhinschi, H. Moncayo, M. Napolitano, J. Davis, and M. Fravolini, "In-flight actuator failure detection and identification for a reduced size UAV using the artificial immune system approach," in *AIAA Guidance, Navigation, and Control Conference*, 2009, p.6266. [Online]. Available: https://doi.org/10.2514/6.2009-6266.

[62] B. Lyons, H. Moncayo, A. Noriega, I. Moguel, and M. G. Perhinschi, "Hardware-in-the-loop Simulation of an extended non-linear dynamic inversion augmented with an immunity-based adaptive control system," in *AIAA Modeling and Simulation Technologies (MST) Conference*, 2013, p.5152. [Online]. Available: doi: 10.2514/6.2013-5152.

[63] H. Moncayo, M. Perhinschi, B. Wilburn, J. Wilburn, and O. Karas, "UAV adaptive control laws using non-linear dynamic inversion augmented with an immunity-based mechanism," in *AIAA Guidance, Navigation, and Control Conference*, 2012, p.4678. [Online]. Available: doi: 10.2514/6.2012-4678.

[64] S. Ozcelik and S. Sukumaran, "Implementation of an artificial immune system on a mobile robot," *Procedia Computer Science,* vol.6, pp.317-322, 2011. [Online]. Available: https://doi.org/10.1016/j.procs.2011.08.058.

[65] B. Hofmann-Wellenhof, H. Lichtenegger, and E. Wasle, *GNSS–global navigation satellite systems: GPS, GLONASS, Galileo, and more*. Springer Science & Business Media, 2007.

[66] P. Roberts, R. Walker, and P. O'Shea, "Fixed wing UAV navigation and control through integrated GNSS and vision," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2005, p.5867. [Online]. Available: https://core.ac.uk/download/pdf/10874623.pdf.

[67] A. Cho *et al.*, "Fully automatic taxiing, takeoff and landing of a UAV using a single-antenna GPS receiver only," in *Control, Automation and Systems, 2007. ICCAS'07. International Conference on*, 2007: IEEE, pp.821-825. [Online]. Available: doi: 10.1109/ICCAS.2007.4407014.

[68] S. M. Sánchez-Naranjo *et al.*, "GNSS vulnerabilities," in *Multi-Technology Positioning*: Springer, 2017, pp.55-77. [Online]. Available: doi: 10.1007/978-3-319-50427-8_4.

[69] Á. A. Da Silva, W. Yamaguti, H. K. Kuga, and C. C. J. M. P. i. E. Celestino, "Assessment of the ionospheric and tropospheric effects in location errors of data collection platforms in equatorial region during high and low solar activity periods," vol.2012, 2012. [Online]. Available: doi: 10.1155/2012/734280.

[70] S. Kaul. (2016) GNSS And Its Vulnerability. *Space Safety Magazine*. [Online]. Available: http://www.spacesafetymagazine.com/magazine/gnss-and-its-vulnerability/.

[71] T. Kos, M. Botinčan, and A. J. P. Dlesk, "Mitigating GNSS positioning errors due to atmospheric signal delays," vol.23, pp.495-513, 2009.

[72] H. V. Sudarshan, *Seamless Sky*, 1st ed. Aldershot, UK: Ashgate Publishing Limited, 2003.

[73] R. D. Easton and P. Muse, *GPS Declassified: From Smart Bombs to Smartphones*. Lincoln, NE: University of Nebraska Press, 2013.

[74] J. Gross and T. E. Humphreys, "GNSS spoofing, jamming, and multipath interference classification using a maximum-likelihood multi-tap multipath estimator," in *Proceedings of the 2017 International Technical Meeting of The Institute of Navigation, Monterey, CA, USA*, 2017, pp.662-670. [Online]. Available: https://doi.org/10.33012/2017.14919.

[75] J. Hardy, *Sensitivity analysis of a relative navigation solution for unmanned aerial vehicles in a GNSS-denied environment*. Morgantown, WV: West Virginia University, 2016.

[76] S. Du, W. Sun, and Y. Gao, "MEMS IMU error mitigation using rotation modulation technique," *Sensors,* vol.16, no.12, p.2017, 2016. [Online]. Available: https://dx.doi.org/10.3390/s16122017.

[77] T. Lou, H. Fu, Z. Wang, and Y. Zhang, "Schmidt-Kalman filter for navigation biases mitigation during mars entry," *Journal of Aerospace Engineering,* vol.28, no.4,

p.04014101, 2015. [Online]. Available: https://dx.doi.org/10.1061/(asce)as.1943-5525.0000423.

[78] M. B. Rhudy, Y. Gu, H. Chao, and J. N. Gross, "Unmanned aerial vehicle navigation using wide-field optical flow and inertial sensors," *Journal of Robotics,* vol.2015, pp.1-12, 2015. [Online]. Available: https://dx.doi.org/10.1155/2015/251379.

[79] B. Etkin, *Dynamics of Atmospheric Flight* (no. Book, Whole). New York, NY: Dover Publications, 2005.

[80] T. R. Kane and D. A. Levinson, *Dynamics, theory and applications*. Ithaca, NY: McGraw Hill, 1985.

[81] R. Meier, S. Bauer, J. Slotboom, R. Wiest, and M. J. M. B. T. S. Reyes, "A hybrid model for multimodal brain tumor segmentation," vol.31, pp.31-37, 2013. [Online]. Available: https://hal.inria.fr/hal-00912934/PDF/proc_brats_2013.pdf#page=35.

[82] M. G. Perhinschi, B. Wilburn, J. Wilburn, H. Moncayo, and O. Karas, "Simulation environment for UAV fault tolerant autonomous control laws development," *Journal of Modeling, Simulation, Identification, and Control,* vol.1, no.4, pp.164-195, 2013. [Online]. Available: doi: 10.7726/jmsic.2013.1011.

[83] M. Alnuaimi, M. Perhinschi, and G. Al-Sinbol, "Immunity-based Framework for Autonomous Flight in GPS-denied Environment," *The International Review of Aerospace Engineering,* Submitted 2018.

[84] M. Alnuaimi and M. Perhinschi, "Immunity approach for UAV position and velocity estimation in GPS-denied environment," in *2018 ISA@Montreal Symposium*, Montreal, Canada, Y. P. Balajee, Ed. 2018: The International Society of Automation.

[85] F. Van Diggelen and P. Enge, "The worlds first GPS MOOC and worldwide laboratory using smartphones," in *Proceedings of the 28th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2015)*, 2015, pp.361-369.

[86] Federal Aviation Administration, "Global Positioning System (GPS) standard positioning service (SPS) performance analysis report," ed. Washington, DC: Washington: Federal Aviation Administration, 2014.

[87] R. M. Rogers, *Applied mathematics in integrated navigation systems*. American Institute of Aeronautics and Astronautics, 2007.

[88] P. Batista, C. Silvestre, P. Oliveira, and B. Cardeira, "Accelerometer calibration and dynamic bias and gravity estimation: Analysis, design, and experimental evaluation," *IEEE transactions on control systems technology,* vol.19, no.5, pp.1128-1137, 2011. [Online]. Available: doi: 10.1109/TCST.2010.2076321.

[89] V. M. Tereshkov, "An intuitive approach to inertial sensor bias estimation," *International Journal of Navigation and Observation,* vol.2013, 2013. [Online]. Available: http://dx.doi.org/10.1155/2013/762758.

[90] W. Gracey, "Measurement of aircraft speed and altitude," National Aeronautics and Space Administration Hampton, VA, Langley Research Center, 1980.